

KIINTEISTÖMASSAN HALLINTAOHJELMISTO

Pasi Revonmäki

Opinnäytetyö
Tekniikan ja liikenteen ala
Insinööri AMK

2015

Tekniikan ja liikenteen ala
Insinööri AMK

Tietokantaohjelmiston luominen käyttöliittymineen Posion kunnan kiinteistömässan hallintaan

Tekijä	Pasi Revonmäki	Vuosi	2015
Ohjaaja	Kenneth Karlsson		
Toimeksiantaja	Posion kunta		
Työn nimi	Kiinteistömässan hallintaohjelmisto		
Sivu- ja liitemäärä	102 + 1		

Opinnäytetyön tarkoituksena oli luoda Posion kunnalle tietokantaohjelmisto, jolla voidaan tietoteknisesti hallinnoida kunnan omistamaa kiinteistömässaa. Hallinnointi tarkoittaa tässä kiinteistöjen eri tietojen vientiä / muokkaamista / talentamista luotavaan tietokantaan helposti käytettävän käyttöliittymän avulla. Kunnalla ei ollut lainkaan keskitettyä tietovarastoa kiinteistömässastaan.

Tietokantoja käyttävät ohjelmistot ovat tunnetusti erittäin kalliita ja useimmiten vieläpä hankalia käyttää johtuen mm. niiden skaalautuvuudesta isoihinkin ympäristöihin, jolloin toimintoja on turhan runsaasti käytettävyyden kustannuksella. Tietokantaohjelmistoksi ja ohjelmointikieliksi valittiin Microsoftin tuotteet, koska niiden käyttö on luontevaa ja koska kunnan muu IT-ympäristö on myös toteutettu pitkälti tämän yrityksen tuotteilla. Lisäksi ohjelmointipuolella lopputyön tekijällä oli aiempaa kokemusta sen käytöstä.

Lopputuloksena saatiinkin halutunlainen tietokanta ja sen käyttämiseen tarkoitettu erillinen käyttöliittymä, jonka käyttäjähallinta perustuu AD-oikeuksiin ja on helposti hallittavissa ja ylläpidettävissä. Tilaa oli tähän toteutukseen tyytyväinen ja koska osaaminen on samassa organisaatiossa, niin jatkotyöstäminen ja ylläpito ovat helppoja toteuttaa. Ohjelmistolla voi olla pienimuotoista kysyntää laajemminkin. Edellytyksenä on että käytännön toteutusta on testailtu riittävässä määrin jotta loppukäyttäjillä on antaa perinpohjaisempaa palautetta.

Asiasanat: Posion kunta, tietokannat, käyttöliittymät, oppimispäiväkirjat

Degree Programme in Information
Technology
Technology, Communication and
Transport

Author	Pasi Revonmäki	Year	2015
Supervisor	Kenneth Karlsson		
Commissioned by	Posio Municipality		
Subject of thesis	Real estate Management Software		
Number of pages	102 + 1		

The purpose of this study was to establish a database software for the municipality of Posio that can manage the real estates owned by the municipality by using information technology. The management here means exporting, modifying and saving data to the database that is to be created by user friendly interface. The municipality did not have any central data storage about real estates owned by the municipality.

The software that uses databases are known to be very expensive and in most cases even difficult to use, due to, among other things, their scalability to large environments, when there are many functions at the expense of usability. Microsoft products are chosen for database software and programming languages, because usage of these are easy and other IT environment of the municipality has also been implemented using Microsoft products. In addition, the author has experience in these programming languages and Microsoft products.

As a final result a desired database and a separate interface to use it were achieved. The interface of the user management was based on the AD rights and the interface is easy to manage and maintain. The customer was satisfied with this creation, and because know-how is in the same organization, so further processing and maintenance are easy to implement. The demand of the software may be restricted, but it can be used in a wider area around the country. It requires that the practical implementation is tested in sufficient quantities and the end-users are expected to give more precise feedback.

Key words Posio municipality, database, user interface, learning diary

SISÄLLYS

1	JOHDANTO	8
2	TIETOKANTAPALVELIN	9
2.1	Tietokantaohjelmiston alustan valmistelu	9
2.2	Tietokantapalvelimen palvelinohjelmisto ja versio	9
2.3	Asennus, tietokanta ja taulut.....	9
2.4	Tietokantapalvelimen asennus <i>Windows 7</i> -ympäristöön	12
2.5	Hallintaohjelmisto <i>Management Studio 2012:n</i> asennus	20
2.6	Tietokantaan yhdistäminen	25
2.7	Tietokantapalvelimen konfigurointi ja verkkojakaminen	28
3	CLIENTIN LUOMINEN JA YHDISTÄMINEN	32
3.1	<i>XP</i> -työaseman/ <i>VB.NET</i> -projektin verkkokäyttö	32
3.2	Clientin yhdistäminen tietokantaan	35
3.3	Moduulin lisääminen projektiin.....	43
4	TIETOKANNAN TAULUT	45
4.1	Alkumäärittelyt ja tarvittavat taulut	45
4.2	Luominen ja ominaisuudet.....	45
4.2.1	Kannan kuvaaja.....	45
4.2.2	Päätaulun luominen.....	46
4.3	TableAdapterit	46
5	TIETOJEN VIEMINEN JA TUOMINEN	52
5.1	Lomakepohjan suunnittelu	52
5.2	BindingNavigator -ohjausobjektin luominen ja konfigurointi	60
6	KÄYTTÖLIITTYMÄ	63
6.1	Päärakenteesta päättäminen.....	63
6.2	Multiform -rakenteen luominen.....	63
6.3	Tiedot DataGridView issä	71
6.4	Alitaulujen hallintalomakkeen käyttöliittymän suunnittelu.....	74
6.5	Päätaulun syöttölomakkeen käyttöliittymän suunnittelu.....	80
6.6	Virheilmoitusten suunnittelu	88
6.7	ToolTip ien lisääminen.....	89
7	RAPORTIT, YHTEENVEDOT JA HYÖDYNTÄMINEN	91

8 TULOSTUSTARPEET JA TULOSTUSMODUULIEN KOODAAMINEN.....	92
9 JOHTOPÄÄTÖKSET	98
LÄHTEET.....	99
LIITTEET	102

ALKUSANAT

Haluan kiittää työnantajaani Posion kuntaa mahdollisuudesta luoda jotain uutta ja hyödyllistä. On aina monin verroin mielekkäämpää tehdä ”real deal” kuin vain simuloida jotain. Samoin tahdon kiittää silloista teknisen toimen johtajaa Jani Aholaa alkuideasta ja avusta aloituksessa.

Kiitollinen olen myös silloiselle esimiehelleni hallintojohtaja Heli Knutarsille mahdollisuudesta käyttää työaikaan tähän projektiin. Isona apuna olivat, kiitos heille, myös rakennusmestarit Ossi Karjalainen ja Jorma Varanka. Heiltä sain termit ja tarpeet kohdalleen. Myös rakennustarkastaja Mikko Ruokamo antoi hyviä ohjeita valikkorakenteisiin, kiitokset myös Mikolle.

Viimeisenä, muttei vähäisempänä tahdon mainita tuutorini Kenneth Karlssonin, jonka apu kaikessa oli korvaamaton. Todella loistavaa palvelua. Kiitos Kenu!

KÄYTETYT MERKIT JA LYHENTEET

Client = asiakas, eli tässä opponentti palvelimelle, eli loppukäyttäjän ohjelma

Events = tapahtumat, tilanne jonka yhteydessä suoritettava subi ajetaan

"Formi" = tietolomake

"Kanta" = tietokanta

"Rimpsu" = merkkijono

"Subi" = aliohjelma

XP = Windows XP, Microsoftin omistama Windows -käyttöjärjestelmä

1 JOHDANTO

Huomasin keväällä 2011 Lapin Kansassa jutun, jossa mainittiin uusi käytännön-läheisempi tapa hankkia alempi korkeakoulututkinto tietotekniikan alalta RAM-Kista. Jutun perusteella se oli kuin suunniteltu itselleni. Siinä kerrottiin opintojen painottavan julkishallinnon tarpeita sekä valmiuksia yrityselämään. Toisaalta minähän olen työssä kunnassa hoitaen samalla omaa tietotekniikkayritystä sivutoimisesti.

Olin jo aiemmin kiinnostunut kehittämään osaamistani tietokantojen osalta, koska vaikka työssäni ja yrityksessäni tein kaikkea tietotekniikkaan liittyvää laidasta laitaan, niin tietokannat olivat alue johon olin vähemmän tutustunut, mutta jolle ajattelin olevan tarvetta.

Kävin siis kuntamme osastopäälliköiltä kysymässä, olisiko heillä mielessään mitään, missä kunta tarvitsisi uutta tietokantaa ja liittymää siihen. Tässä vaiheessa teknisen toimen osastonjohtaja toi esille kunnan tarpeen tällaisesta ohjelmistosta, johon vietäisiin kunnan kiinteistömassan tiedot. Pidettyämme alustavan palaverin asiasta totesimme ajatuksen olevan toteuttamiskelpoinen.

2 TIETOKANTAPALVELIN

2.1 Tietokantaohjelmiston alustan valmistelu

Kun projekti alun perin vuonna 2013 aloitettiin, niin kunnan työasemat olivat *XP*:llä ja silloin valittiin vanhemmat ohjelmistot. Siis koska tietokantapalvelin oli lisenssi- yms. syistä normaali työasema, niin sen käyttöjärjestelmä oli alkujaan *Windows XP 32 -bit PRO FI*, kuten kunnan muutkin työasemat silloin.

Siihen tehtiin normaali perus *XP*-asennus kuten muillekin kunnan työasemille siihen aikaan. Sovellusohjelmistot jätettiin asentamatta. Katsottiin kuitenkin, että keskusmuistia oli riittävästi palvelinohjelmiston pyörittämiseen.

2.2 Tietokantapalvelimen palvelinohjelmisto ja versio

Tietokantatuotteeksi valittiin *Microsoftin SQL Server*, koska se toimii parhaiten *Microsoftin AD* -ympäristössä ja *VB.NET*:llä koodatuissa clienteissa. Tietokantapalvelimeksi tuli *MS SQL Server 2008 R2*, koska se tuki kiertoteitse vielä *XP*:tä.

Kun projekti aloitettiin uudelleen 2015, niin työasemat olivat siirtyneet *Windows 7* -kantaan, eli niissä oli käyttöjärjestelmänä *Windows 7 64 -bit PRO FI*. Tästä syystä tietokantapalvelin vaihdettiin eri tietokoneeseen ja valittiin versioksi *MS SQL Server 2012 SP1*.

2.3 Asennus, tietokanta ja taulut

Projekti aloitettiin marraskuussa 2013. Asennettu versio oli *MS SQL Server 2008 R2*. Seuraavassa on päiväkirja siitä, kuinka suoritin alkuperäisen asennuksen.

Latasin ensin ohjelman asennuspaketit:

<http://www.microsoft.com/en-us/download/details.aspx?id=30438>

Päivitin *XP*:n asentajan:

<http://www.microsoft.com/fi-fi/download/details.aspx?id=8483>

Asensin ohjeen mukaan:

<http://blogs.msdn.com/b/petersad/archive/2009/11/13/how-to-install-sql-server-2008-r2-express-edition-november-ctp.aspx>

Määritin tilin ja sen salasanan kohdassa 7.

Määritin tilin ja sen salasanan kohdassa 8.

Asensin *Management Studion*

<https://www.microsoft.com/en-us/download/details.aspx?id=7593>

Asensin *Microsoft Visual Studio 2010 Expressin*

<http://www.microsoft.com/visualstudio/eng/products/visual-studio-2010-express>

Tämä vaati poikkeavaa ratkaisua, koska *MS* ei enää jaa pakettia, vaan sivu ohjautuu/pakottaa versioiden *2012* ja *2013* äärelle. Estämällä sivun uudelleenohjaus onnistuu ladata tuo vanhempi versio. Päivitin paketit.

Ryhdyin luomaan tietokantaa näillä ohjeilla:

<http://msdn.microsoft.com/en-us/library/ms233763%28v=vs.90%29.aspx>

Aloitin *VB.NET*:lla uuden projektin, joka luo tuon haluamani tietokannan ja antaa työkalut sen rakenteen muokkaamiseen.

Annoin tietokannalle nimeksi ***KiinteistomassaKanta***

Ääkkösiä olen oppinut välttelemään tietokanta-asioiden kanssa.

Lisäilin tauluja. Tein paikallisen kannan. Siirsin kannan fyysisesti kantapalvelimessa oikeaan sijaintiin. Testasin toimivuuden. Ryhdyin tutkimaan kuinka otan yhteyden kantaan etänä.

Ensin tutkin serverin verkkoasetuksia:

[http://technet.microsoft.com/en-us/library/ms189083\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms189083(v=sql.105).aspx)

Serverikoneesta piti löytyä *Sql Server Configuration Manager* ja asiakaskoneista *SQL Server Native Client*

Latauksen tein täältä:

<http://www.microsoft.com/en-us/download/details.aspx?id=27596>

Otin serverissä auki ohjelman *SQL Server Configuration Manager* ja aktivoin protokollat sekä käynnistin palvelun uudelleen.

[http://technet.microsoft.com/en-us/library/ms345343\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms345343(v=sql.105).aspx)

[http://technet.microsoft.com/en-us/library/ms181087\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms181087(v=sql.105).aspx)

Tarvitsin tietokannalle käyttäjätunnuksen, jonka tein seuraavasti:

<http://technet.microsoft.com/en-us/library/aa337545.aspx>

<http://msdn.microsoft.com/en-us/library/aa337562.aspx>

Suunnittelin jo mielessäni alustavasti kuinka automatisoin tarvittavien ohjelmien asennuksen työasemiin. Tulen luomaan asennuspaketin tarvittavilla ohjelmilla, joihin minulla on jo kokemusta.

2.4 Tietokantapalvelimen asennus *Windows 7* -ympäristöön

Sitten kuluikin aikaa Posion kunnan sisäisen myllerryksen vuoksi (mm. tekninen johtaja erosi jo vuosi toista takaperin ja uusi valittu vasta nyt) toista vuotta ja projekti aloitettiin uudelleen keväällä 2015. Nyt versioksi valittiin *MS SQL Server 2012 SP2 Express*.

Tässä vaiheessa työasemien käyttöjärjestelmiksi oli vaihtunut *Windows 7 64 bit FI* ja siksi päätettiin ottaa serveripuolellakin käyttöön tuoreempi versio eli *MS SQL 2012 SP2 Express*. Asennus aloitettiin nollasta ajatellen, että se yhdistetään sitten lopuksi vanhaan kantaan.

Ensin poistettiin kaikki *MS SQL 2008* data ja lähdettiin asentamaan tuoreempaa versiota ns. puhtaalta pöydältä. Seuraavassa on päiväkirja siitä.

Latasin ensin asennuspaketit (Kuvio 1).

<https://www.microsoft.com/en-us/download/details.aspx?id=29062>

Nimi	Muokauspäiväm...	Tyyppi	Koko
SQLEXPR_x64_ENU.exe	29.4.2015 13:26	Sovellus	303 168 kt
SQLEXPR_x86_ENU.exe	29.4.2015 13:25	Sovellus	268 193 kt
SQLEXPR32_x86_ENU.exe			
SQLEXPRADV_x64_ENU.exe			
SQLEXPRADV_x86_ENU.exe			
SQLEXPRWT_x64_ENU.exe			
SQLEXPRWT_x86_ENU.exe	29.4.2015 13:39	Sovellus	1 122 570 kt
SqlLocalDB.msi	29.4.2015 13:21	Windows Installer ...	35 648 kt
SQLManagementStudio_x64_ENU.exe	29.4.2015 13:37	Sovellus	972 991 kt
SQLManagementStudio_x86_ENU.exe	29.4.2015 13:37	Sovellus	986 437 kt

Tiedoston kuvaus: Microsoft SQL Server 2012 Express SP2
 Yritys: Microsoft Corporation
 Tiedoston versio: 11.0.5058.0
 Luomispäivä: 29.4.2015 13:22
 Koko: 296 Mt

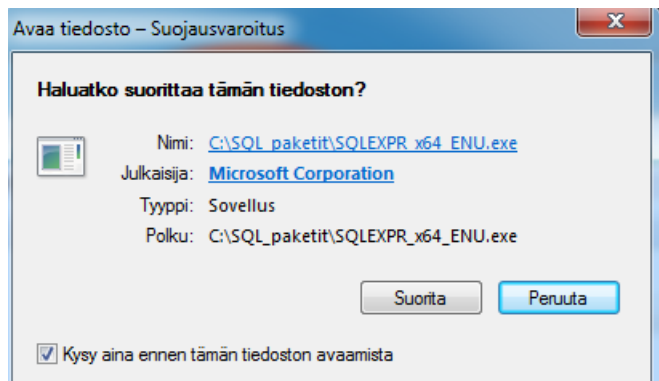
Kuvio 1. Asennuspaketit

Asensin ohjeen mukaan:

<https://technet.microsoft.com/en-us/library/bb500469%28v=sql.110%29.aspx>

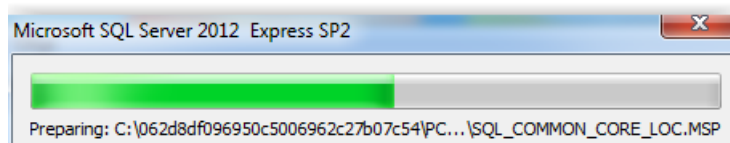
<https://technet.microsoft.com/en-us/library/ms143219%28v=sql.110%29.aspx>

Kaikkia ohjeen kohtia ei tullut esille (liekö johtunut siitä, että versioni *Express* ja taso *SP2*), vaan esittelen ne jotka tuli. Kohtien otsikoiden numerot ovat linkistä löytyvän ohjeen numeroinnin mukaisia. Windows 7 kysyi lupaa asennuksen aloittamiseen ohjeen kohta 1 (Kuvio 2).



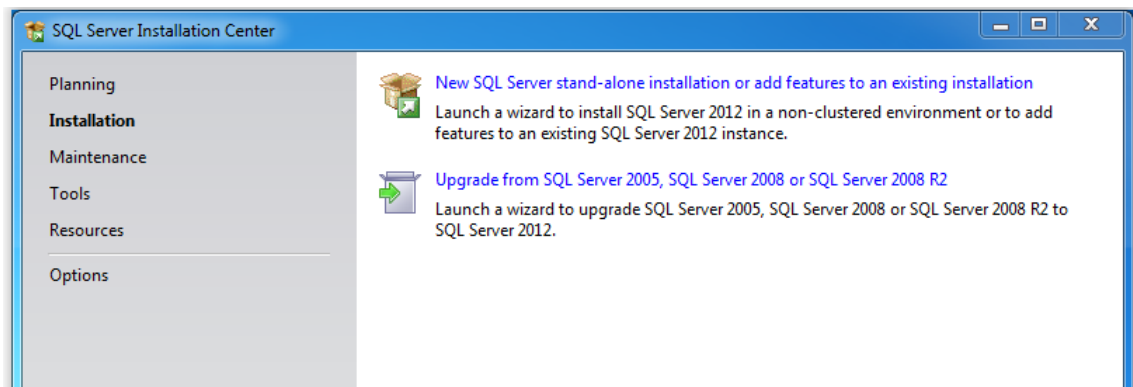
Kuvio 2. **UAC**:n (User Account Control) kysely

Asennus lähti etenemään (Kuvio 3).



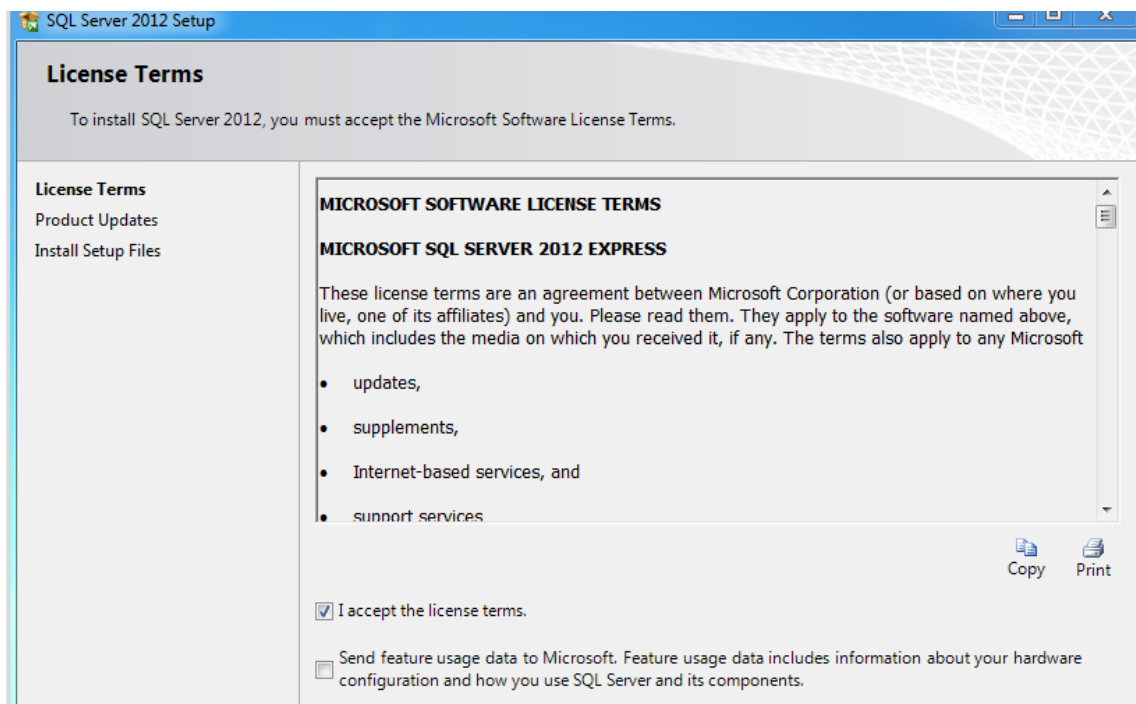
Kuvio 3. Asennuksen eteneminen

2. Valitsin ***New SQL Server stand-alone installation or add features to an existing installation*** (Kuvio 4).



Kuvio 4. Asennustapa

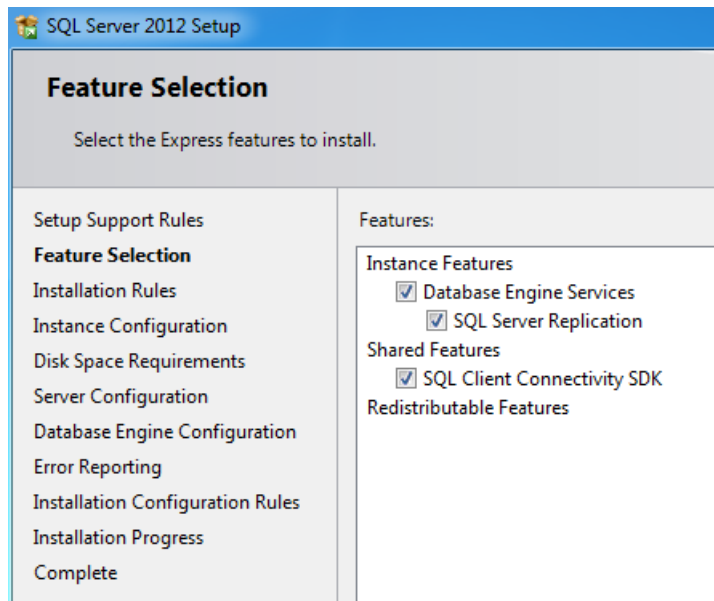
8. Hyväksyin lisenssiehdot (Kuvio 5).



Kuvio 5. Ehtojen hyväksyntä

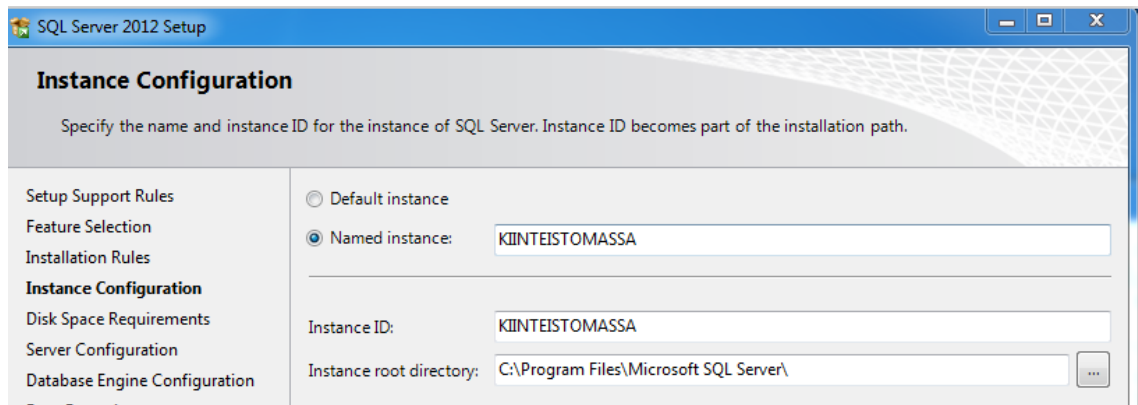
12. Annoin olla kaikki valittuina **Next** ja sitten vain odotin.

Seuraavaksi tuli ominaisuuksien valinta (Kuvio 6).



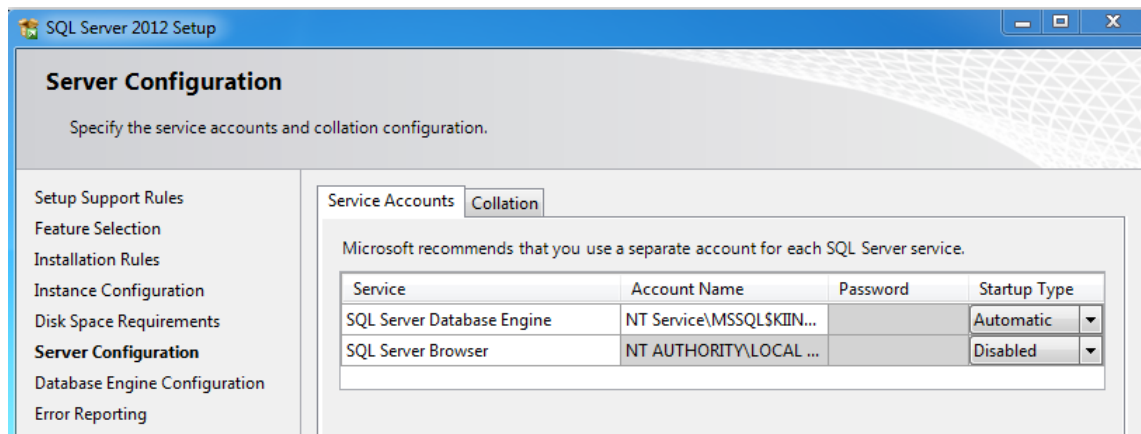
Kuvio 6. Ominaisuuksien valinta

15. Asetin instanssin nimeksi ja ID:ksi **KIINTEISTOMASSA**, asennuspolku ehdotettu, **Next** (Kuvio 7).



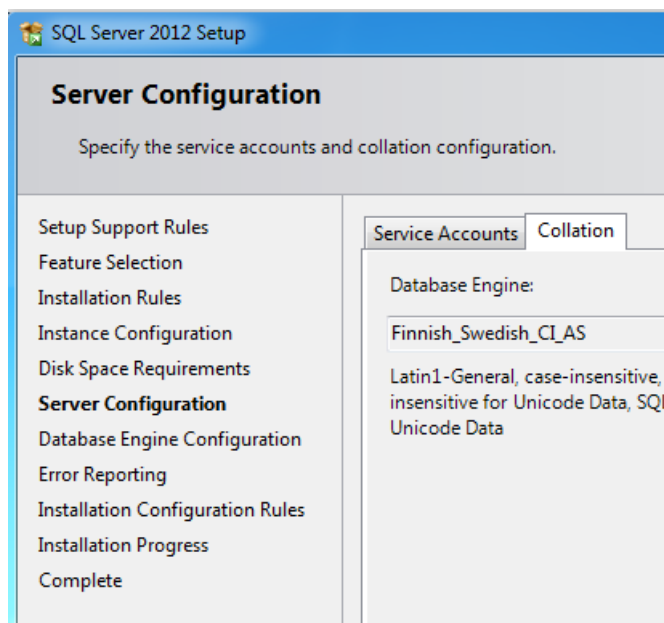
Kuvio 7. Instanssin nimen asettaminen

19. Oletuksilla, **Next** (Kuvio 8).



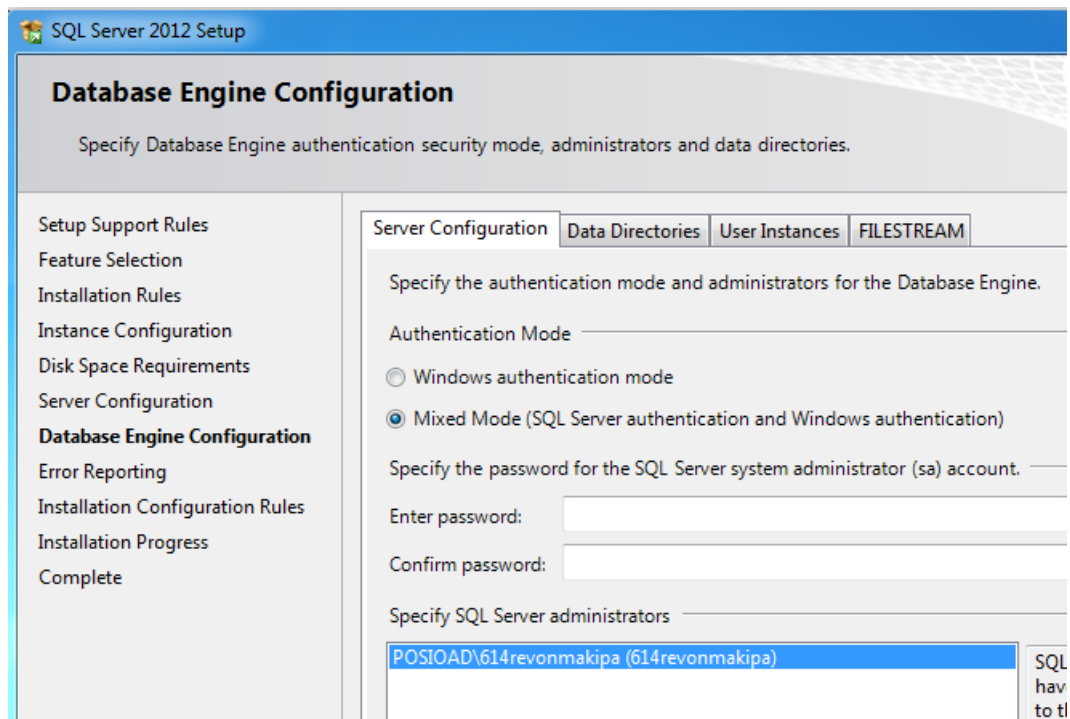
Kuvio 8. Asetukset

Lisää asetuksia (Kuvio 9). **Next.**



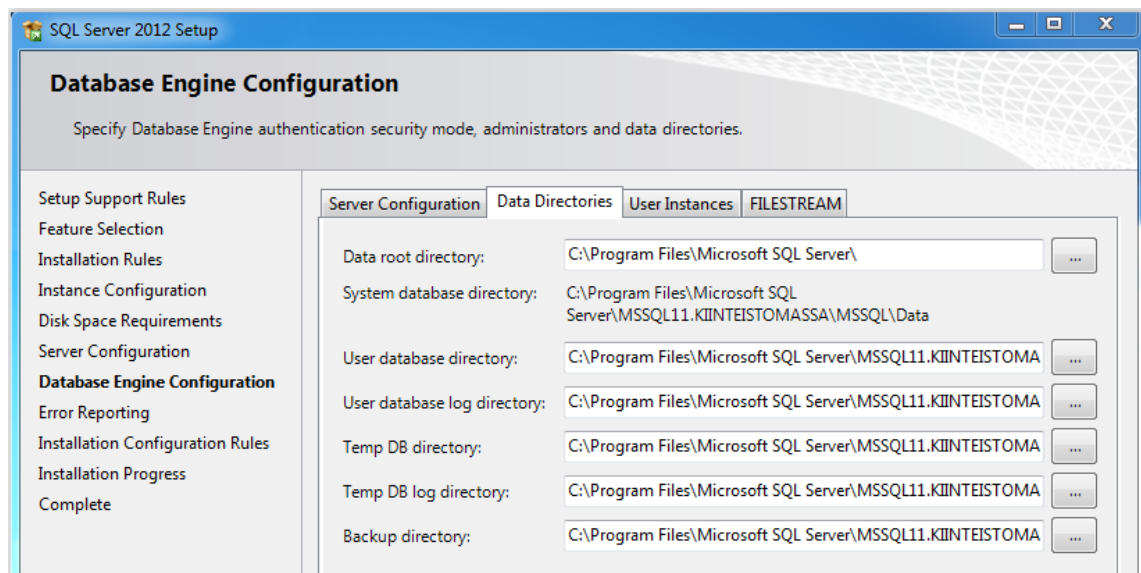
Kuvio 9. Lisää asetuksia

20. Otin **mixed moden**, jotta voin kirjautua kantaan sekä kiinteällä tunnuksella, että toimialuetunnuksella (jolloin voin käyttää toimialueemme toiminnallisuuksia). Kävin asettamassa valinnat päälle **FILESTREAM**-välilehdelle, jotta kantaan saa yhteyden myös etänä. Valitsin moodiksi **Mixed Moden**. *System Admin*-in **sa** salasanaksi laitoin XXXXXXXXX (Kuvio 10).



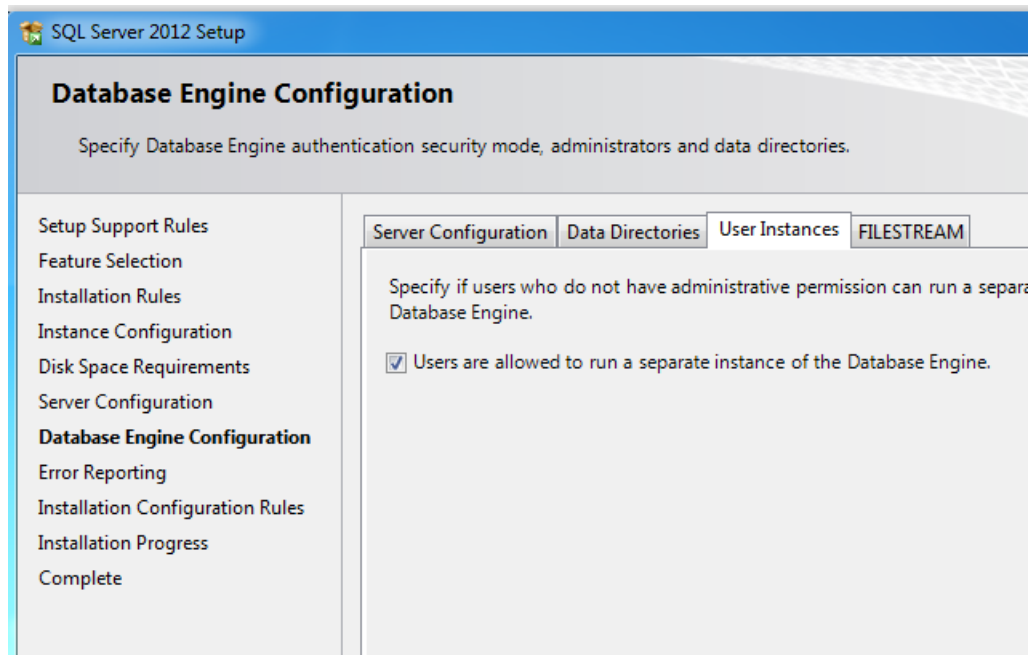
Kuvio 10. Kirjautumistavan valinta

Kansioasetuksien (Kuvio 11) annoin olla oletusarvoilla.



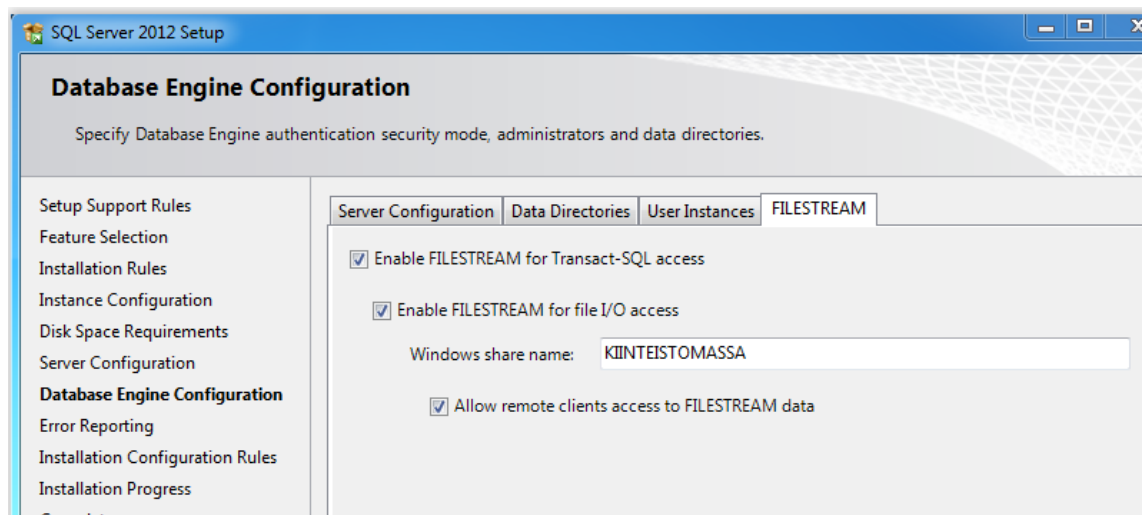
Kuvio 11. Kansioasetukset

Käyttäjien instanssien annoin myös olla oletuksien (Kuvio 12).



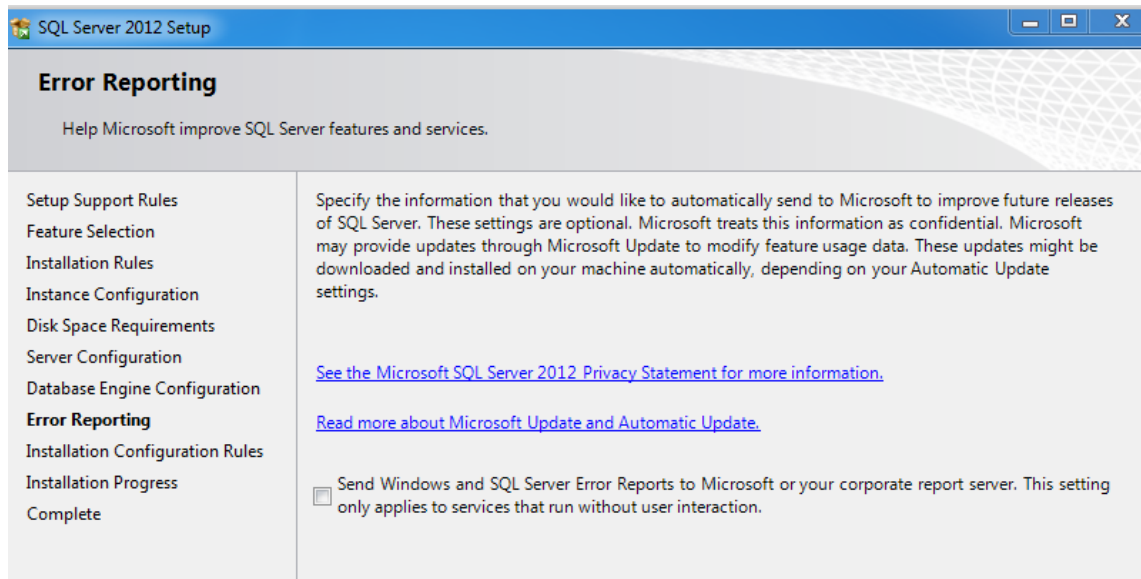
Kuvio 12. Käyttäjien instanssit

Filestreamin otin käyttöön seuraavin asetuksin (Kuvio 13).



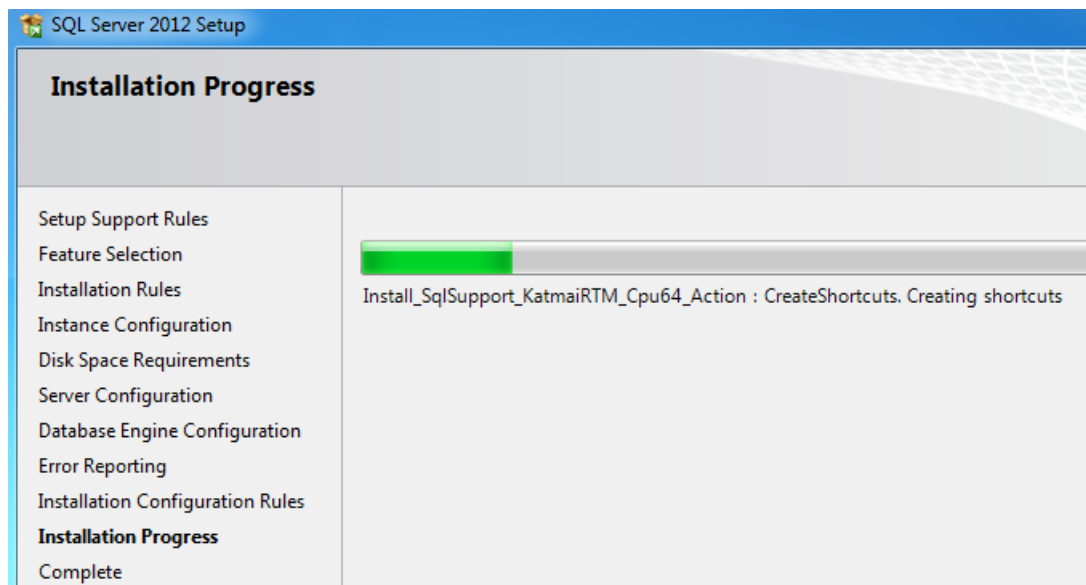
Kuvio 13. Filestreamin asetukset.

Ei virhetietoja MS:lle (Kuvio 14).



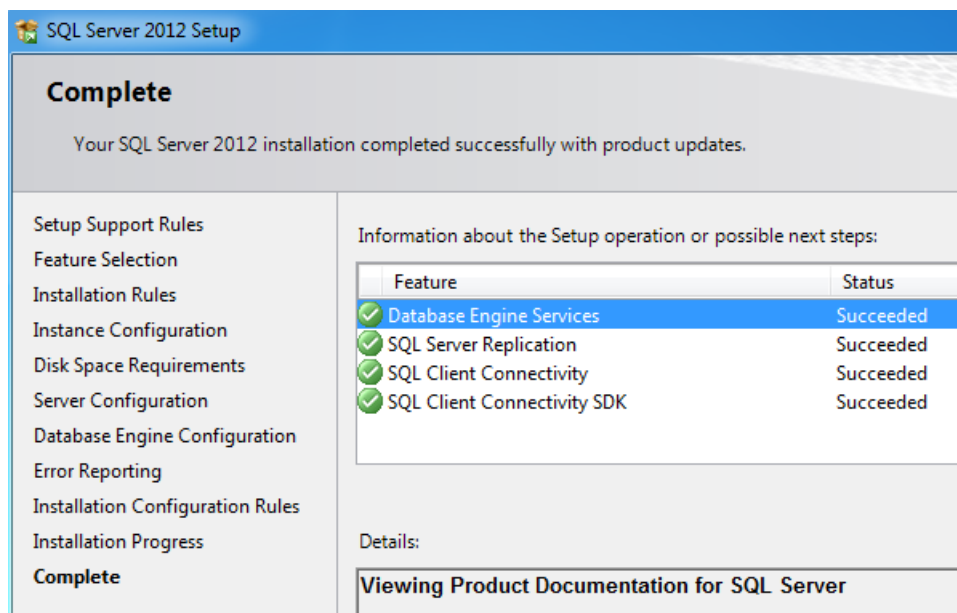
Kuvio 14. Virhetietojen lähettämisesetukset

Asennus lähti etenemään (Kuvio 15).



Kuvio 15. Asennuksen eteneminen

Tuli tieto asennuksen valmistumisesta, **Close** (Kuvio 16). Käynnistin ohjelman uudelleen.



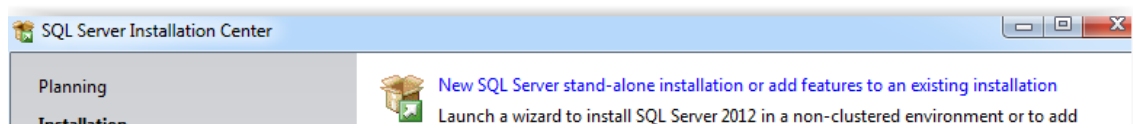
Kuvio 16. Asennuksen valmistuminen

2.5 Hallintaohjelmisto *Management Studio 2012:n* asennus

Asensin *Management Studio 2012:n*, eli tuplaklikkasin 64-bittisen asennuspakettia, jonka latsin täältä:

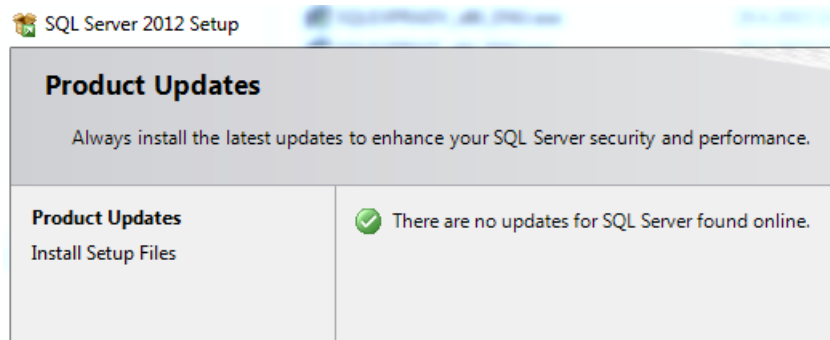
<https://www.microsoft.com/en-us/download/details.aspx?id=29062>

Tulin taas asennuksen jälkeen tuttuun asennusikkunaan, josta valitsin ylimmän vaihtoehdon **New SQL** (Kuvio 17).



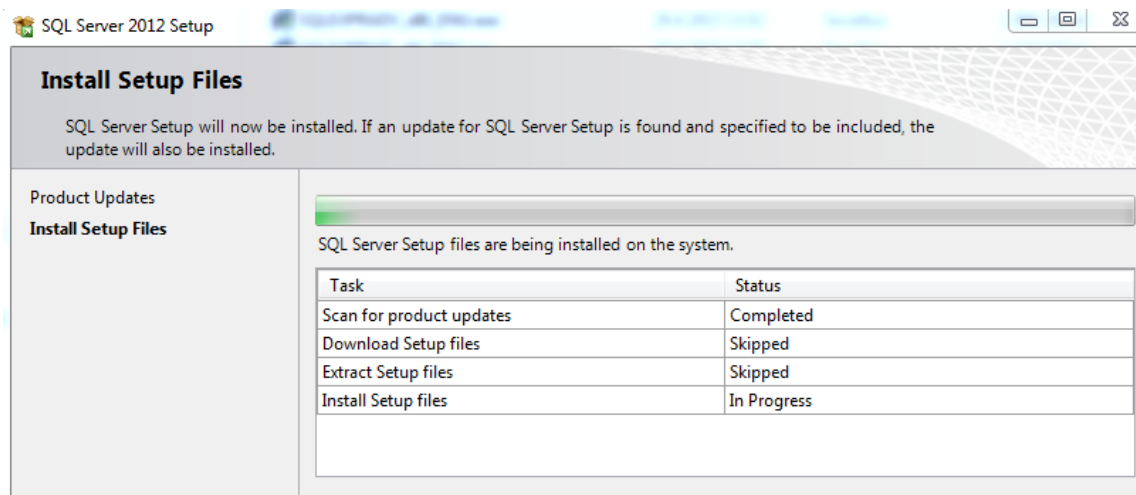
Kuvio 17. Ominaisuuksien lisääminen palvelinohjelmistoon

Ei ollut päivityksiä (Kuvio 18).



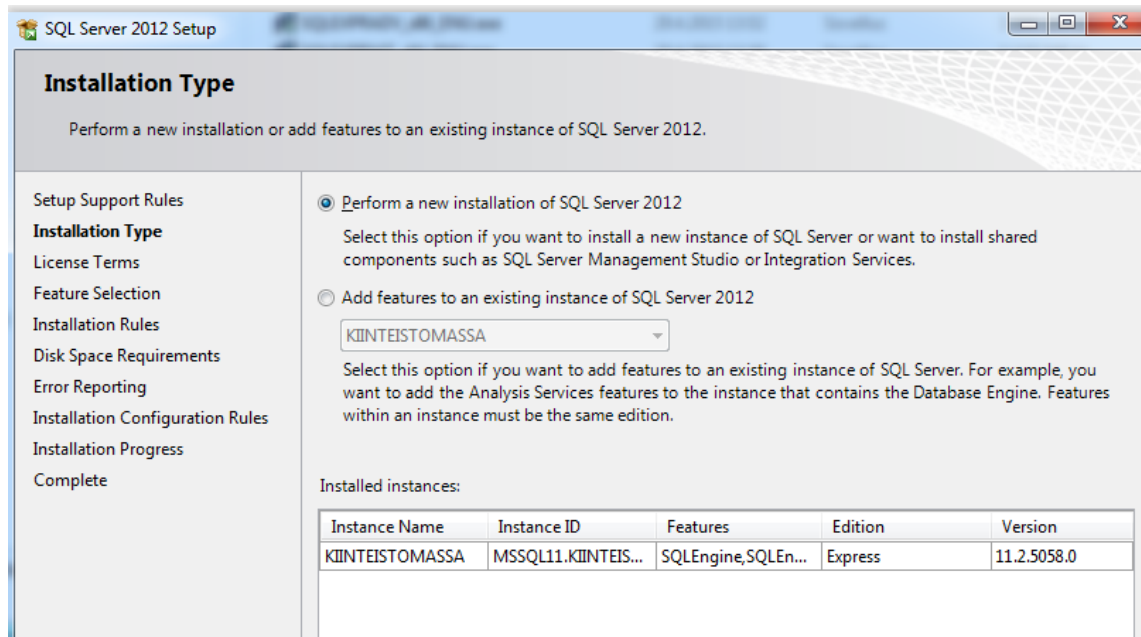
Kuvio 18. Päivitysinfo

Asennus lähti etenemään (Kuvio 19).



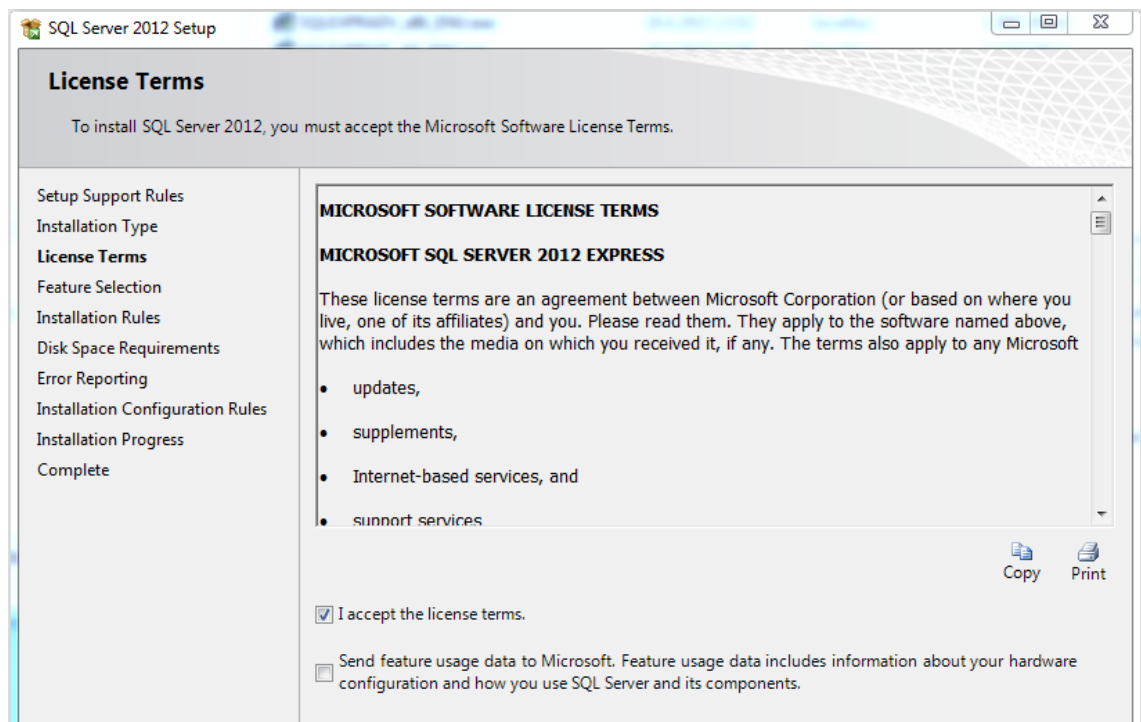
Kuvio 19. Asennuksen eteneminen

Sitten seuraavassa ikkunassa otin ylemmän vaihtoehdon, vaikka voisi olettaa, että ollaan lisäämässä ominaisuuksia (Kuvio 20).



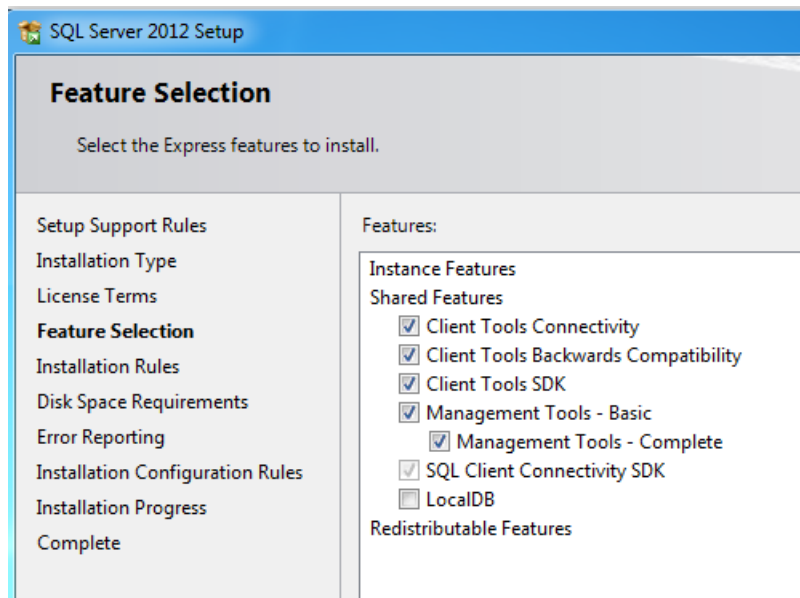
Kuvio 20. Asennustyyppin valinta

Hyväksyin jälleen ehdot (Kuvio 21).



Kuvio 21. Ehtojen hyväksyntä

Sain eteeni ominaisuusvalinnat, joista otin kaikki muut paitsi **LocalDB**:n, koska palvelin ja tietokanta tulevat olemaan käynnissä kaiken aikaa (Kuvio 22).



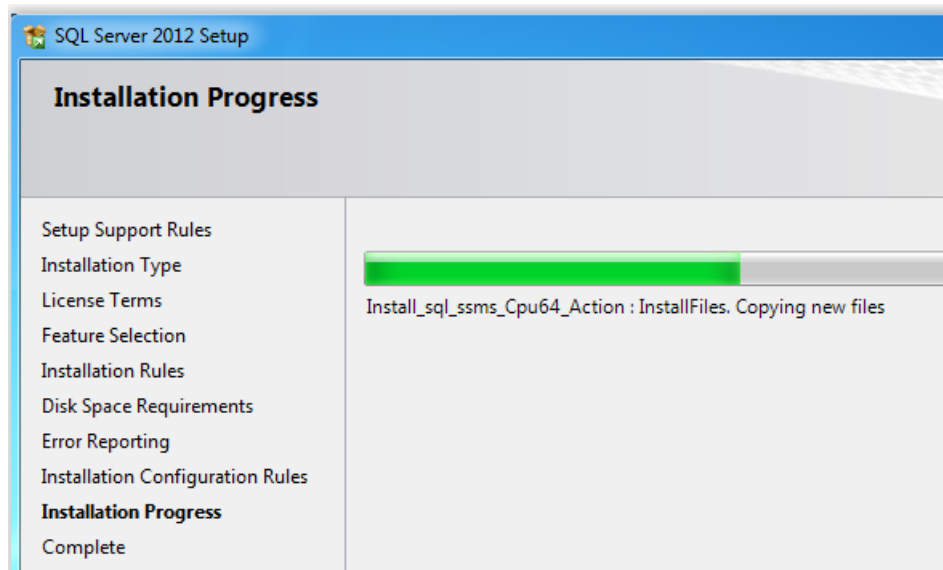
Kuvio 22. Ominaisuuksien valinta

Virhetietojen lähetyksestä MS:lle en aktivoinut tälläkään kertaa (Kuvio 23).



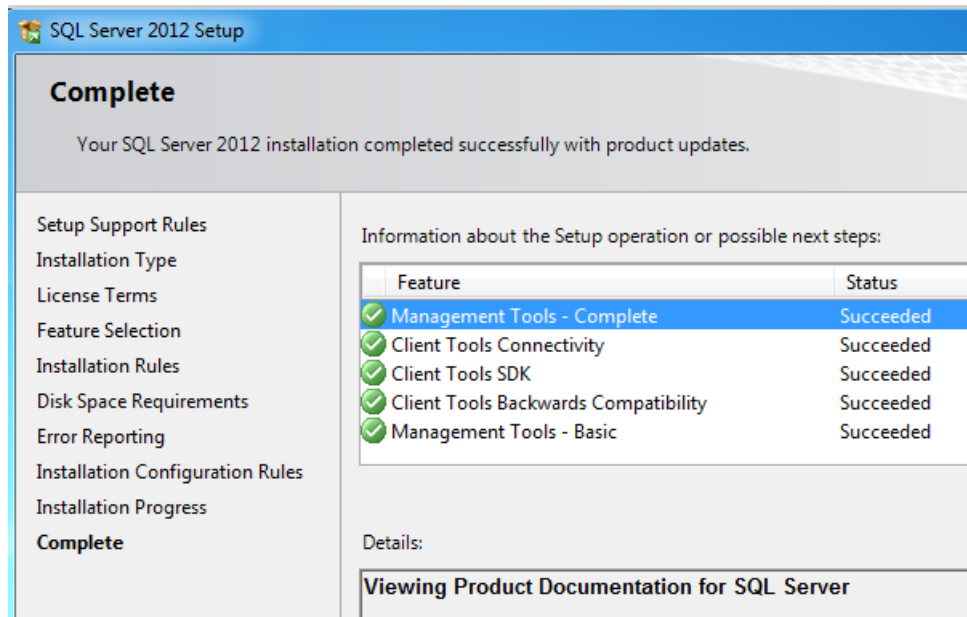
Kuvio 23. Ei lähetetä virhetietoja

Asennuksen eteneminen (Kuvio 24).



Kuvio 24. Asennuksen eteneminen

Asennus suoritettiin onnistuneesti (Kuvio 25).

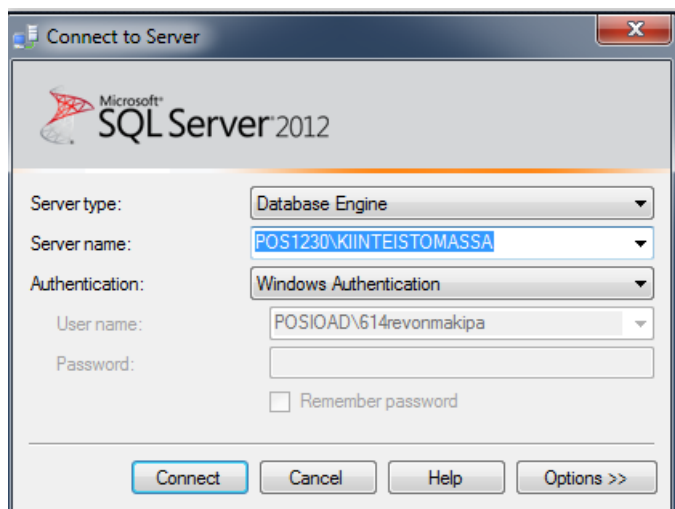


Kuvio 25. Asennuksen valmistuminen

2.6 Tietokantaan yhdistäminen

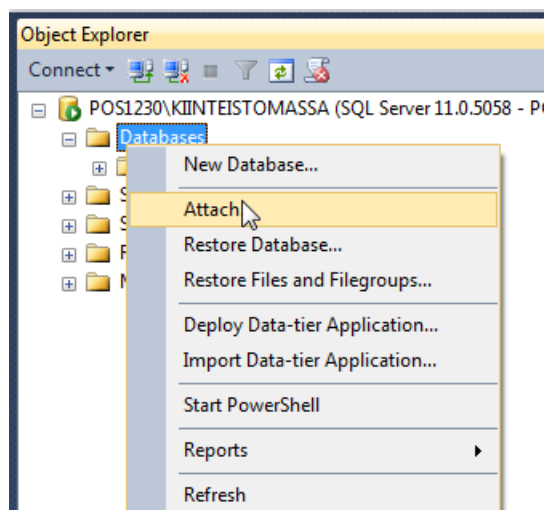
2008 R2:n asennuksen jäljiltä minulla oli jo kanta, johon päätin yrittää ottaa siihen yhteyden nyt asentamalla *Management Studiolla*.

Kirjauduin juuri asentamaani tietokantapalvelimeen toimialuetunnuksellani (Kuvio 26).



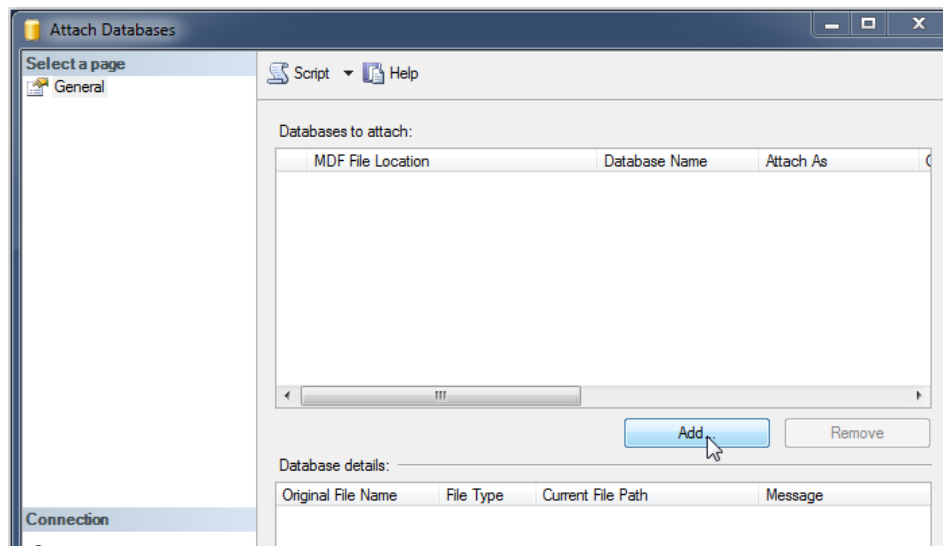
Kuvio 26. Palvelimeen kirjautuminen

Valitsin **Object Explorerissa Database** (tietokanta) -kansion päällä hiiren oikealla näppäimellä **Attachin** (Kuvio 27).



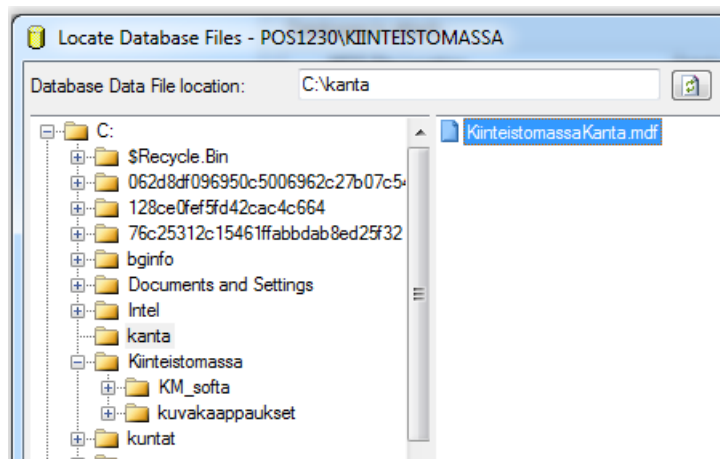
Kuvio 27. Kantaan yhdistäminen

Klikkasin **Add...** (Kuvio 28).



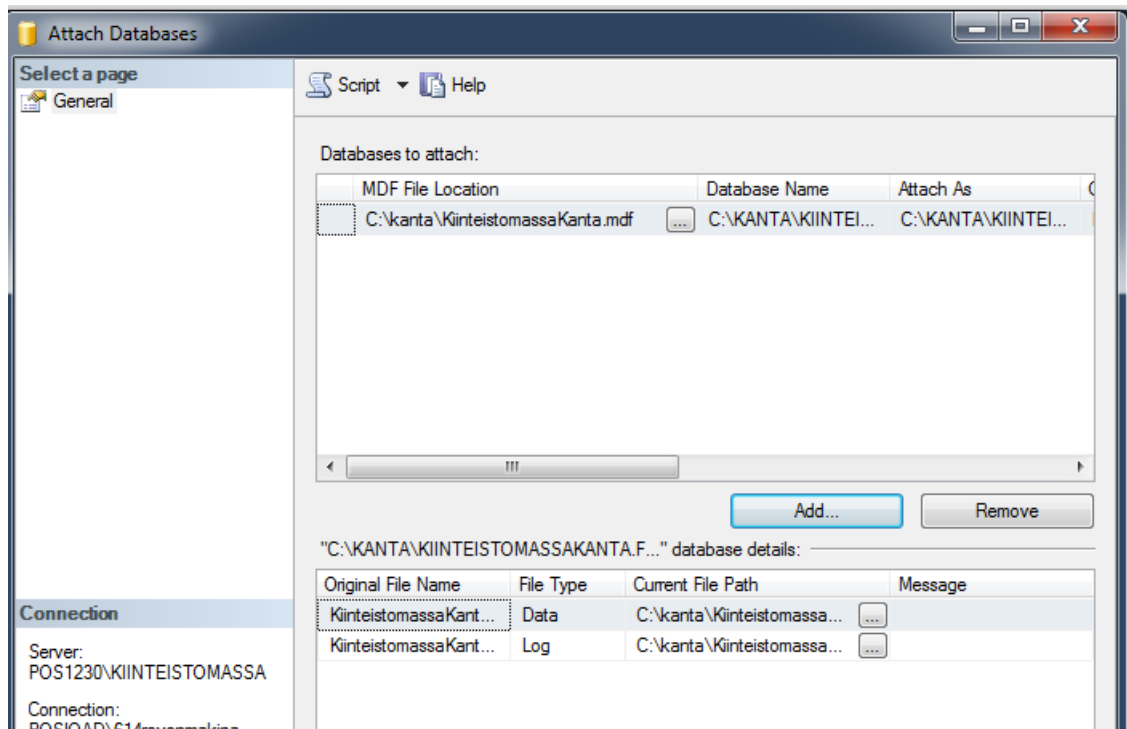
Kuvio 28. Kantaan yhdistäminen

Kävin valitsemassa aiemmin luomani kannan (Kuvio 29).



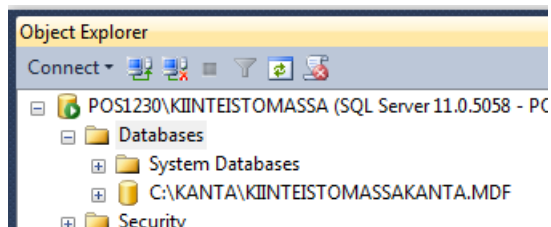
Kuvio 29. Kannan valinta

Kanta oli yhdistetty ja käytettävissä (Kuvio 30).



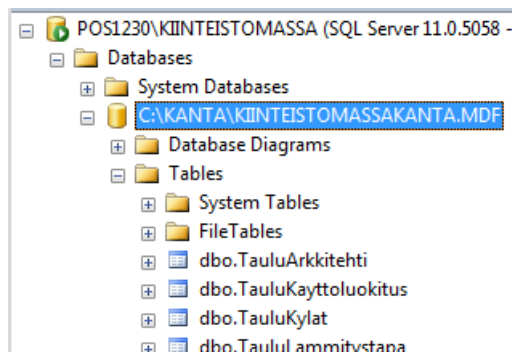
Kuvio 30. Kantayhteys

Kantaan yhdistäminen onnistui, ja kanta ilmestyi listalle (Kuvio 31).



Kuvio 31. Tietokantalistaus

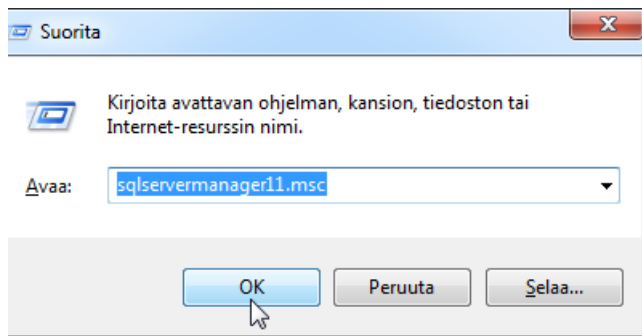
Samoin ilmestyivät aiemmin luomani taulut kansioon **Tables** (Kuvio 32).



Kuvio 32. Taululistaus

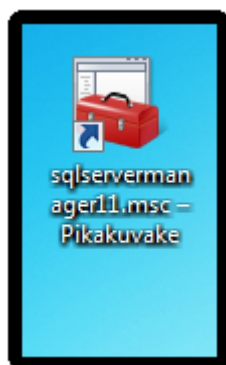
2.7 Tietokantapalvelimen konfigurointi ja verkkojakaminen

Sql Server Configuration Manager oli tässä versiossa automaattisesti mukana, ja sen sai käyntiin komennolla **sqlservermanager11.msc** (Kuvio 33).



Kuvio 33. Sql Server Configuration Managerin käynnistäminen

Tein Sql Server Configuration Managerista pikakuvakkeen serverin työpöydälle (Kuvio 34).

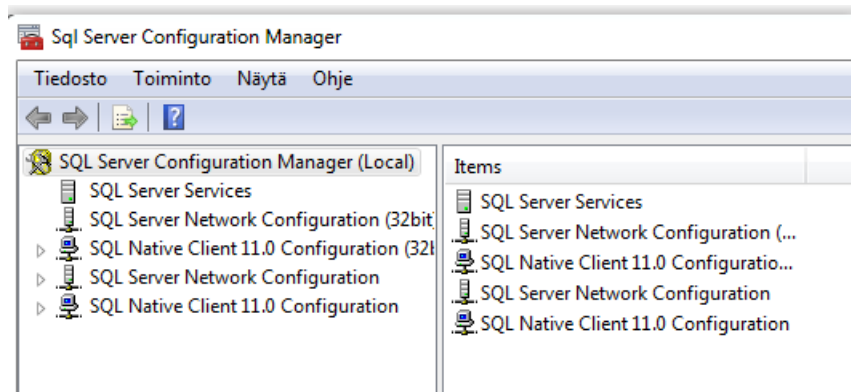


Kuvio 34. Sql Server Configuration Manager -käynnistyskuvake

Seurasin seuraavaa ohjetta:

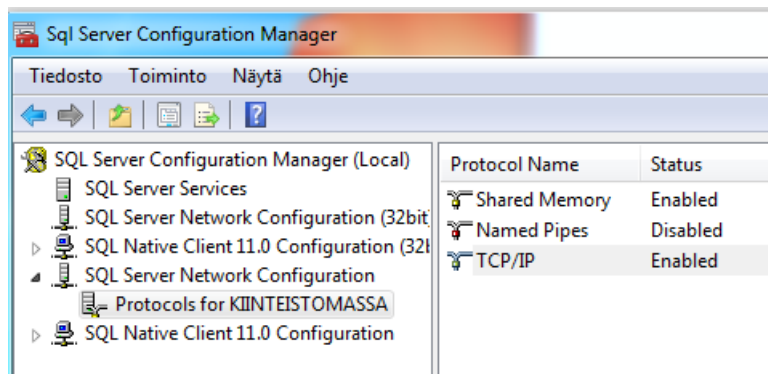
[http://technet.microsoft.com/en-us/library/ms345343\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms345343(v=sql.105).aspx)

Aukaisin serverissä ohjelman *SQL Server Configuration Manager* (Kuvio 35).



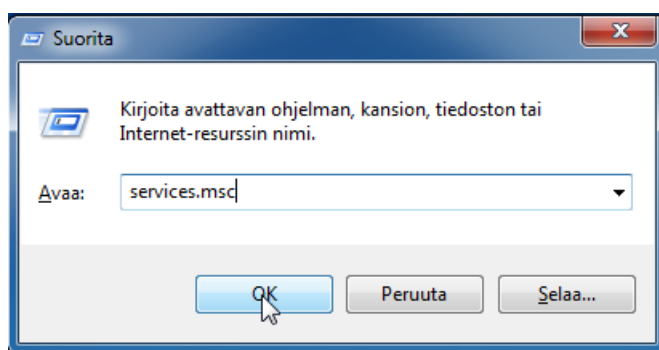
Kuvio 35. Palvelimen määitykset

Aktivoin ensin tarvittavat protokollat (Kuvio 36).



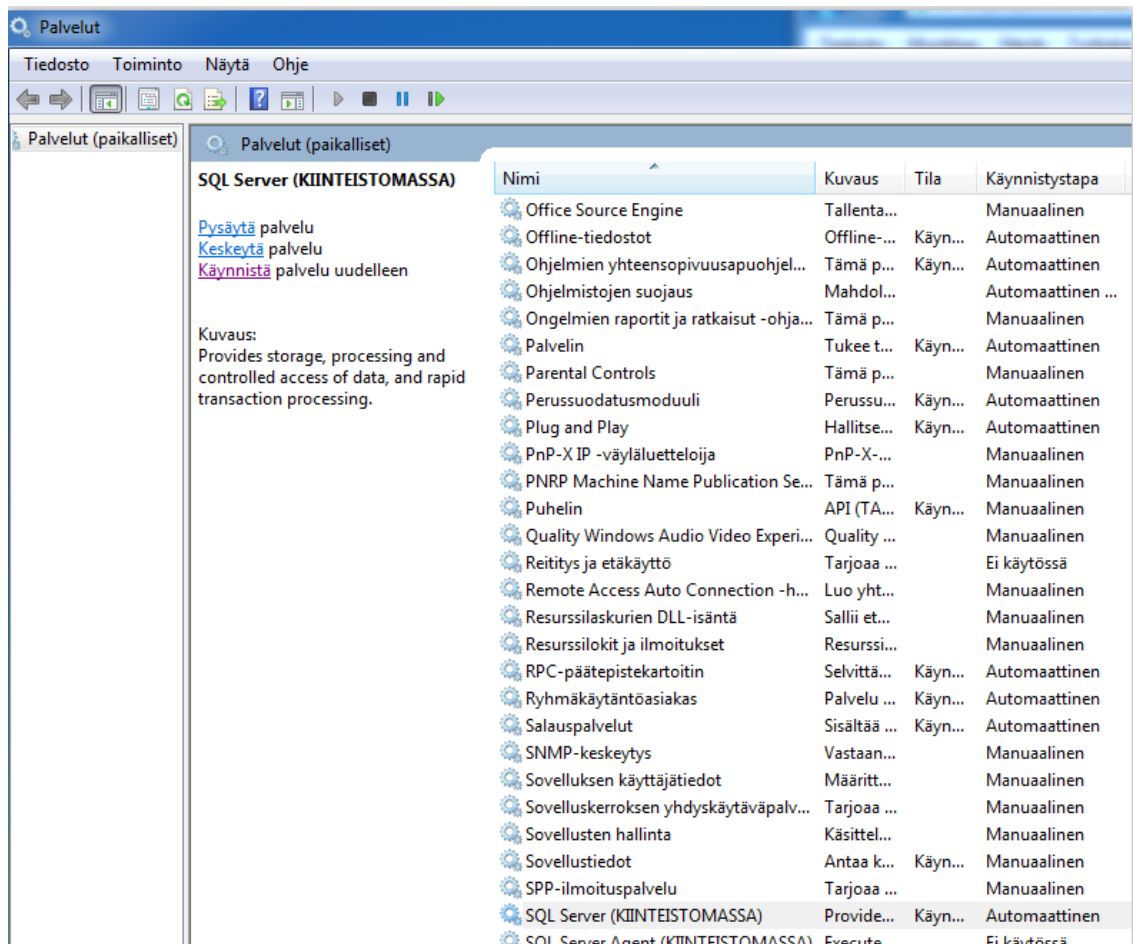
Kuvio 36. Protokollien aktivointi

Tämän jälkeen käynnistin palvelun uudelleen. Palveluihin pääsee näin (Kuvio 37):



Kuvio 37. Palveluihin pääseminen

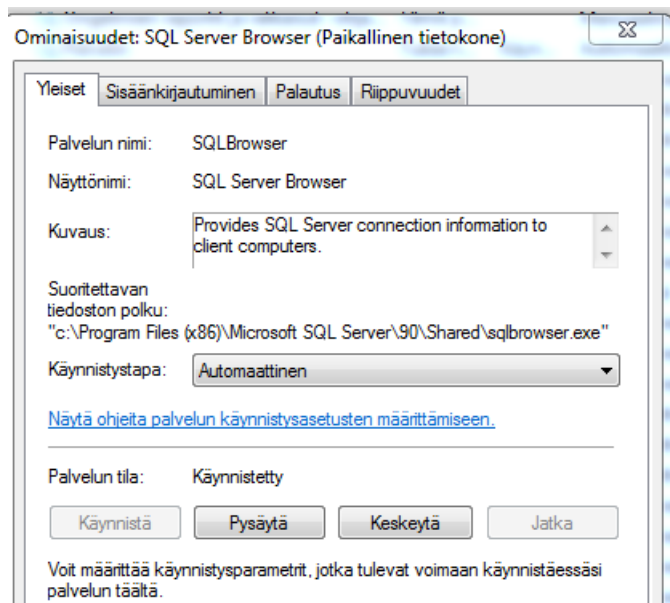
Palveluiden listaus näytti sitten tältä (Kuvio 38):



Kuvio 38. Palveluiden listaus

[http://technet.microsoft.com/en-us/library/ms181087\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms181087(v=sql.105).aspx)

Kävin palveluissa (palveluihin pääsee komennolla **services.msc**) käynnistämässä **SQL Server Browser** -palvelun. Asetin palvelun myös automaattisesti käynnistäväksi (Kuvio 39).



Kuvio 39. SQL Server Browser -palvelun käynnistäminen

Tässä vaiheessa olin päässyt samaan vaiheeseen kuin aiemmin *MS SQL 2008 R2*:n kanssa ja kertosin ensin client-version vanhoista muistiinpanoistani (alla).

3 CLIENTIN LUOMINEN JA YHDISTÄMINEN

3.1 XP -työaseman/VB.NET -projektin verkkokäyttö

Paikalliseen kantaan yhdistäminen olisi mennyt näin:

[http://technet.microsoft.com/en-us/library/ms345332\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms345332(v=sql.105).aspx)

<http://msdn.microsoft.com/en-us/library/vstudio/ms239722.aspx>

Asiakaskoneen verkkoasetukset asetin näin:

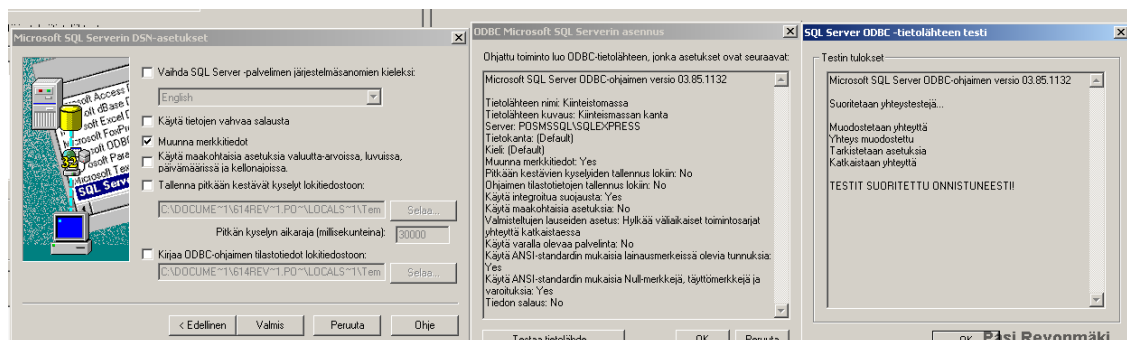
[http://technet.microsoft.com/en-us/library/ms190611\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms190611(v=sql.105).aspx)

Asiakaskoneen tietokantayhteyden lähdin luomaan näin:

[http://technet.microsoft.com/en-us/library/ms188691\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms188691(v=sql.105).aspx)

<http://www.homeandlearn.co.uk/NET/nets12p4.html>

Tietokannan määrittäminen onnistui (Kuvio 40).



Kuvio 40. Tietokannan määrittäminen

Halusin päästä VB.NET:llä kiinni kantaan **ODBC**:n avulla, joten käytin näitä ohjeita:

<http://support.microsoft.com/kb/310985>

Sitten huomasin, että *Express* versio *Visual Studio 2010*:sta on rampautettu juuri tässä kohtaa. Ei-paikallista (verkossa olevaa) kantaa ei siis saakaan näkymään **Database Explorerissa**.

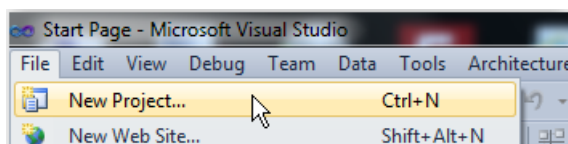
Tätä varten olivat oppilaitoksen tarjoamat työkalut/mahdollisuudet. Otin *Thunderbirdistä* *RAMKin* sähköpostin auki ja pyyhkäisin pölyt sähköpostiin tulleesta postista, jossa kerrottiin oppilaitokset hoitaneen minulle *Dreamspark*-tilin.

Hoidin rekisteröinnin loppuun ja menin katsomaan tarjontaa ja hintoja. Tuorein versio olisi ollut tätä kirjoitettaessa *2013 RC*, mutta, kun en periaatteessa koske *release candidate* versioihin, niin etsin vanhempaa tarjontaa.

Sieltä tilasin latauslinkin *Visual Studio 2010 Ultimateen*. Se on laajin versio sisältäen kaikki ”hienoudet”, eikä *Dreamsparkin* opiskelijatilin kautta hankittuna maksanutkaan mitään:

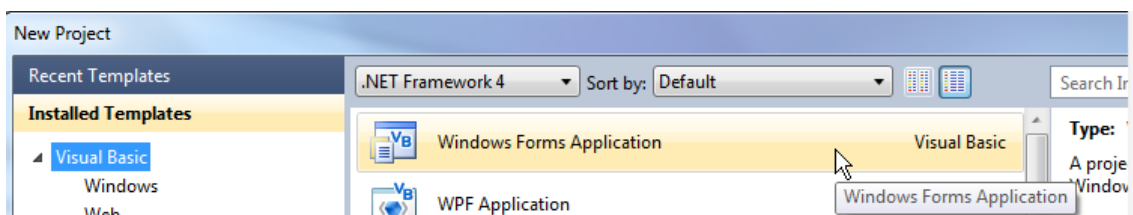
http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#Visual_Studio_Ultimate

Asensin tuon ja aloitin siinä uuden projektin (Kuvio 41).



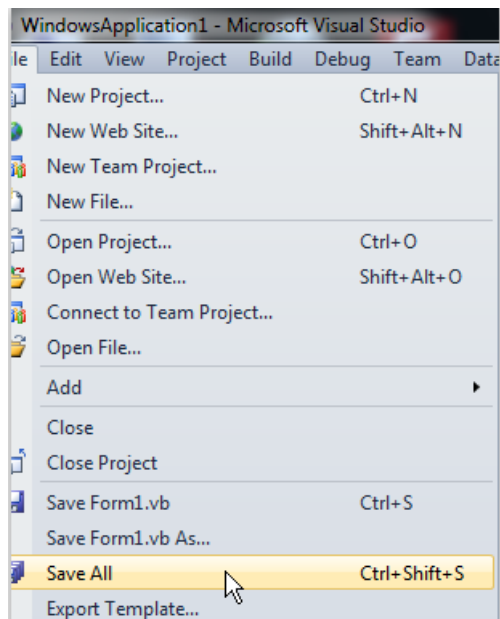
Kuvio 41. Uuden projektin luominen

Valitsin normaalin lomakeprojektin (Kuvio 42).



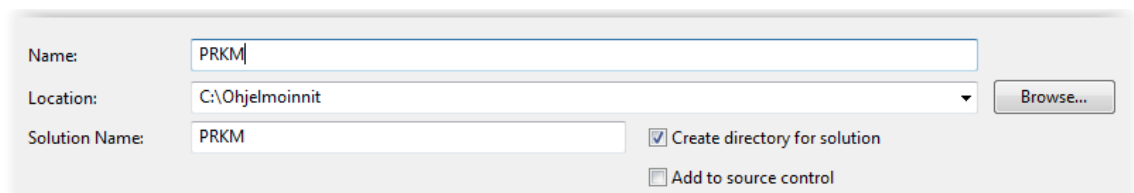
Kuvio 42. Projektin tyypin valinta

Tallensin projektin (Kuvio 43).



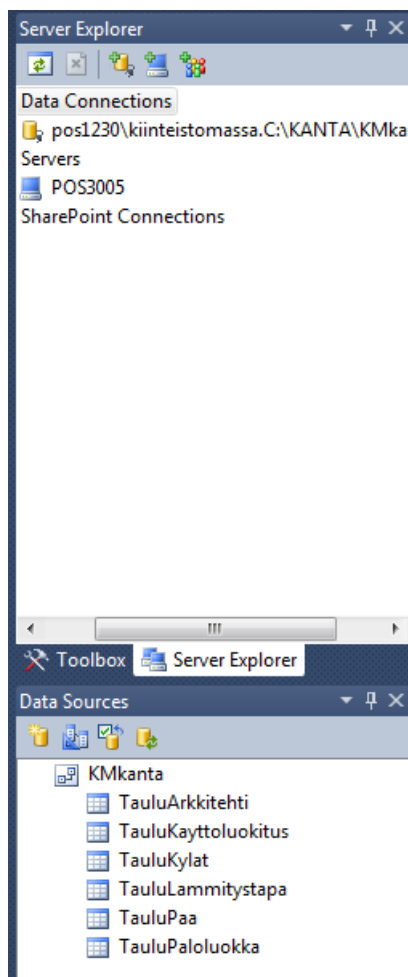
Kuvio 43. Projektin tallentaminen

Nimesin projektin ja vein sen haluamaani sijaintiin (Kuvio 44).



Kuvio 44. Tallennusnimi ja -sijainti

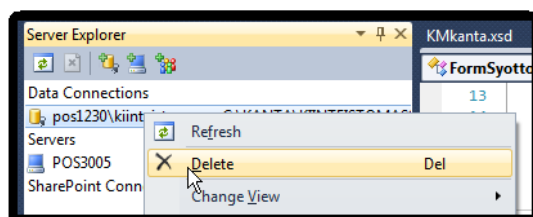
Asetin tietokantatyökalupalkit näkymään (kuva tosin kirjoitushetken tilanteesta) (Kuvio 45).



Kuvio 45. Tietokantatyökalut

3.2 Clientin yhdistäminen tietokantaan

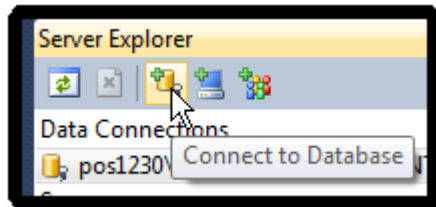
Olin nyt siis saanut kerrattua kantaan yhdistämisen *XP*:llä ja *VS 2010 Ultimatel*-la. Avasin aiemmin *VB.NET*:llä tekemäni projektin Windows 7 -koneessa ja poistin ensin *Server Explorer*issa vanhan tietokantayhteyden. Nyt tein saman *Windows 7 SP1* -työasemalla (Kuvio 46).



Kuvio 46. Tietokantayhteyden poistaminen

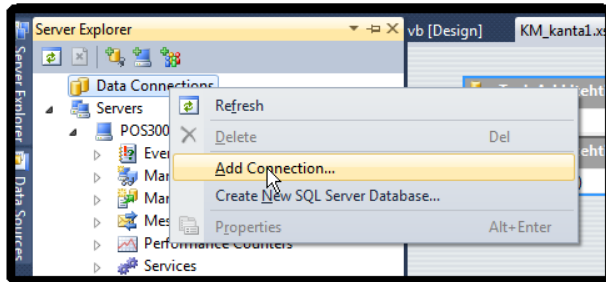
Sitten lähdin luomaan uutta.

Tähän tapaan (Kuvio 47).



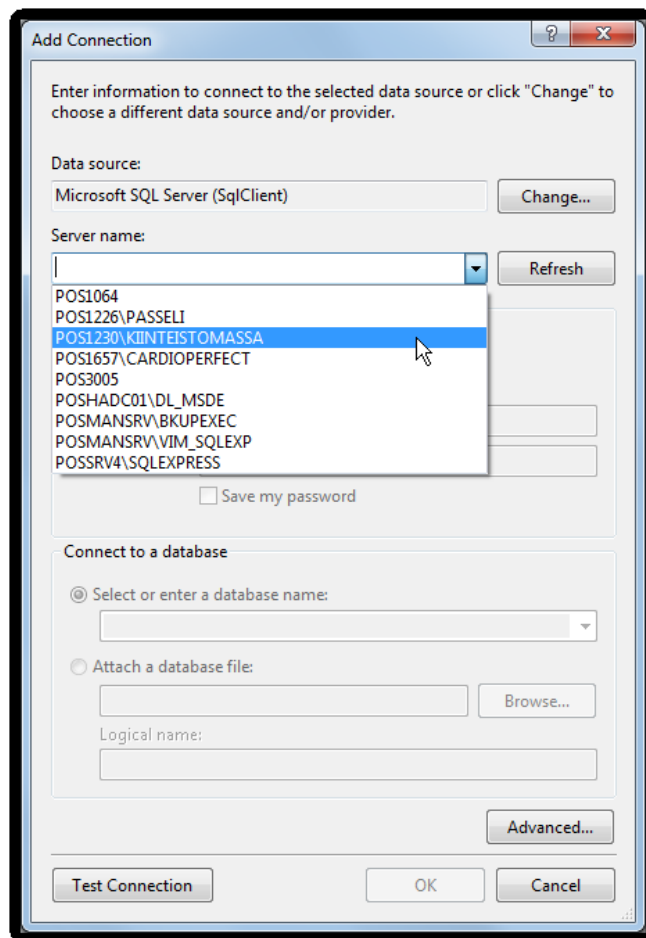
Kuvio 47. Tietokantayhteyden luominen

Vaihtoehtoisesti näin (Kuvio 48):



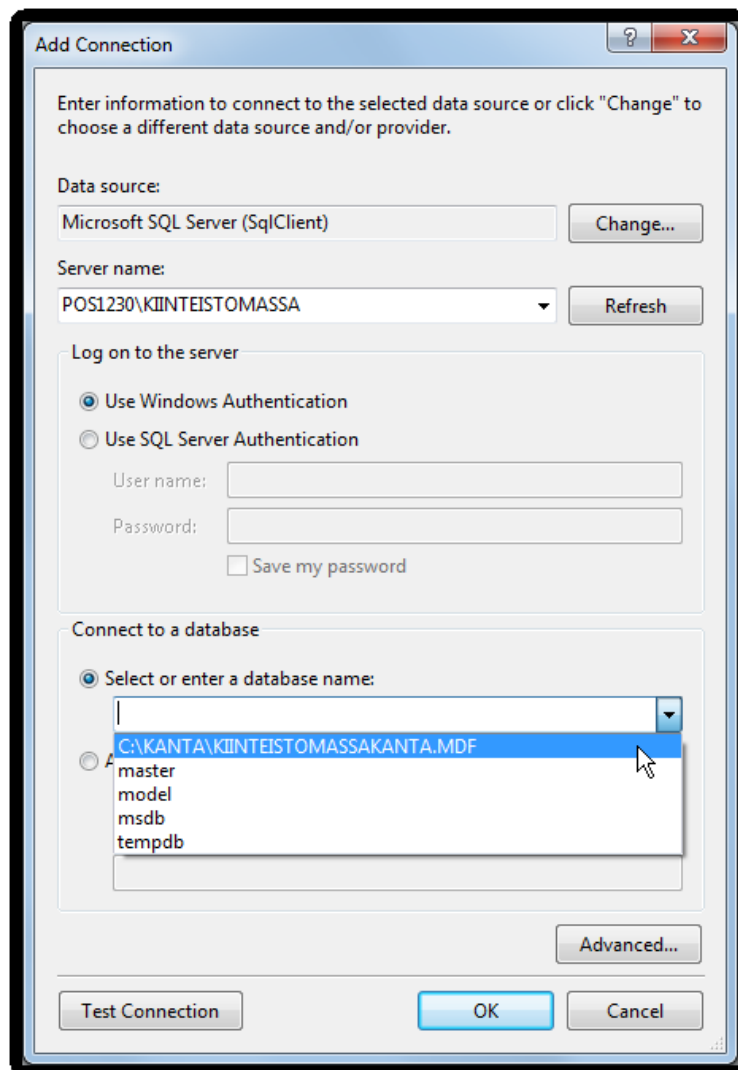
Kuvio 48. Tietokantayhteyden luominen

Valitsin oikean tietokantapalvelimen listasta (Kuvio 49).



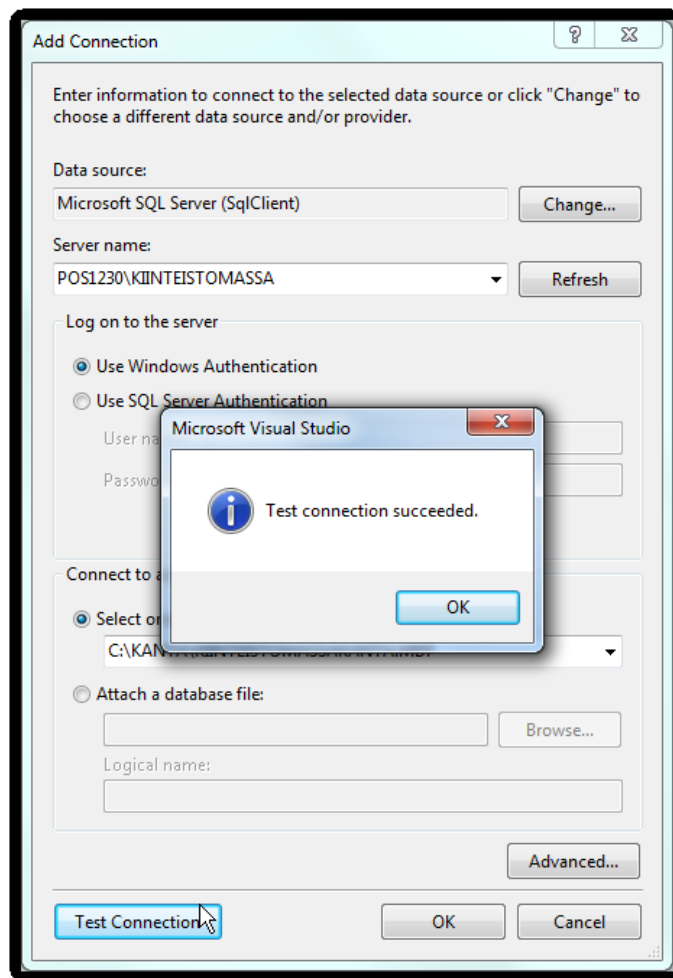
Kuvio 49. Tietokantayhteyden valinta

Valitsin oikean kannan palvelimelta (Kuvio 50).



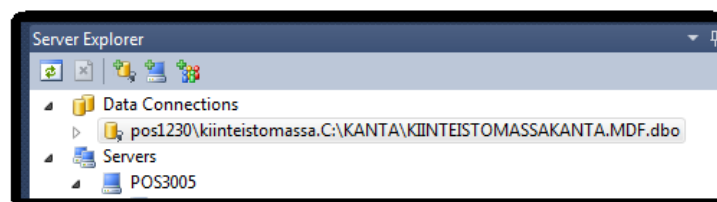
Kuvio 50. Tietokannan valinta

Testasin, että yhteys toimii (Kuvio 51).



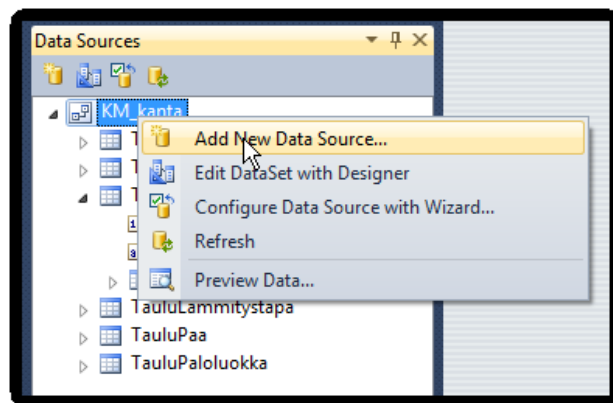
Kuvio 51. Yhteyden testaaminen

Nyt yhteys kantaan oli luotu (Kuvio 52).



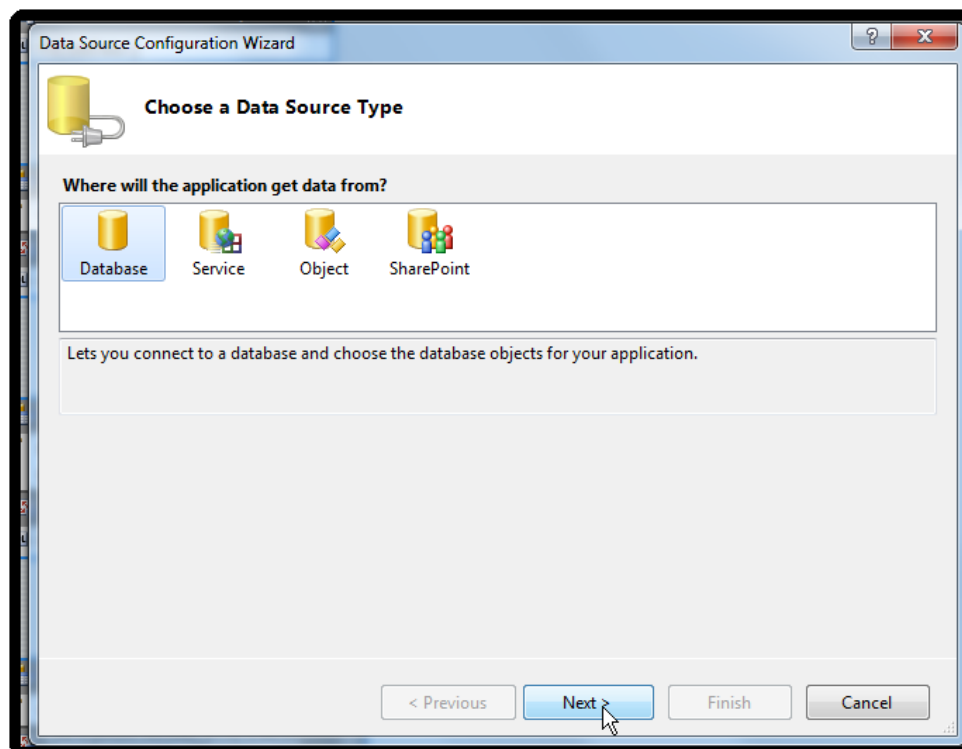
Kuvio 52. Tietokannan näkyminen yhteyksissä

Lisäsin uuden tietolähteen (Kuvio 53).



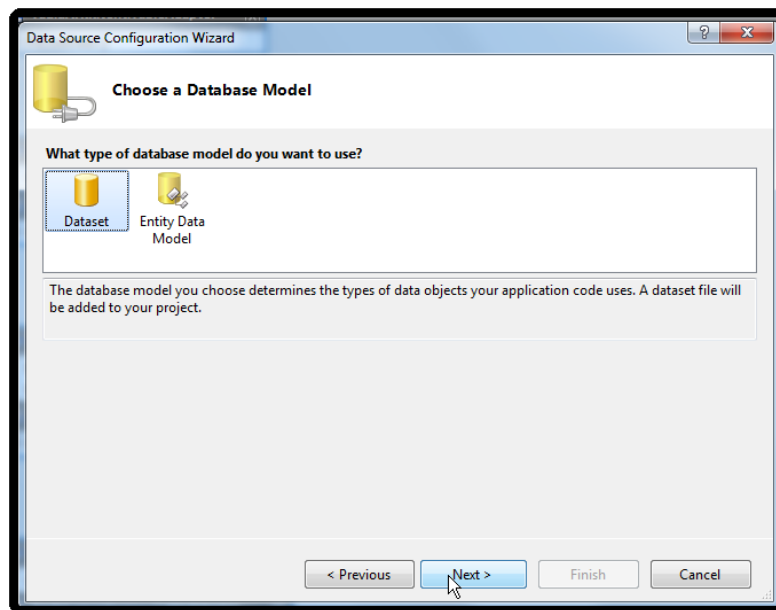
Kuvio 53. Tietolähteen lisääminen

Tietolähteen tyyppi (Kuvio 54).



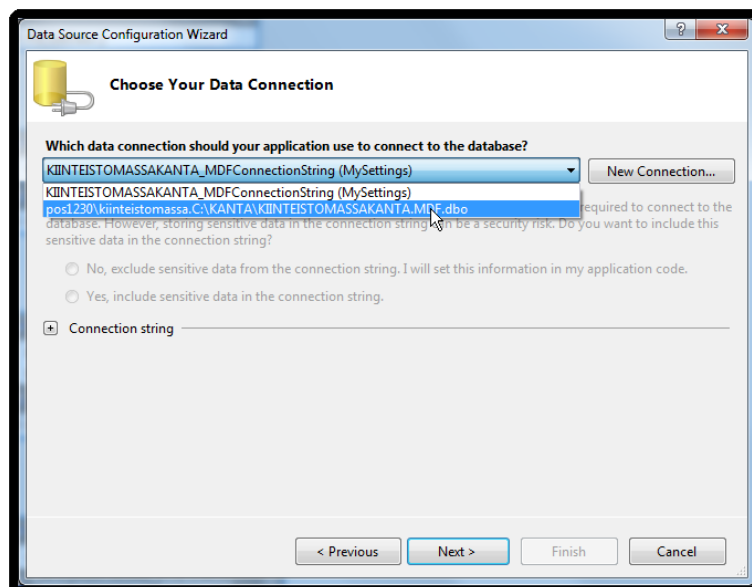
Kuvio 54. Tietolähteen tyyppi

Tietokannan tyyppi (Kuvio 55).



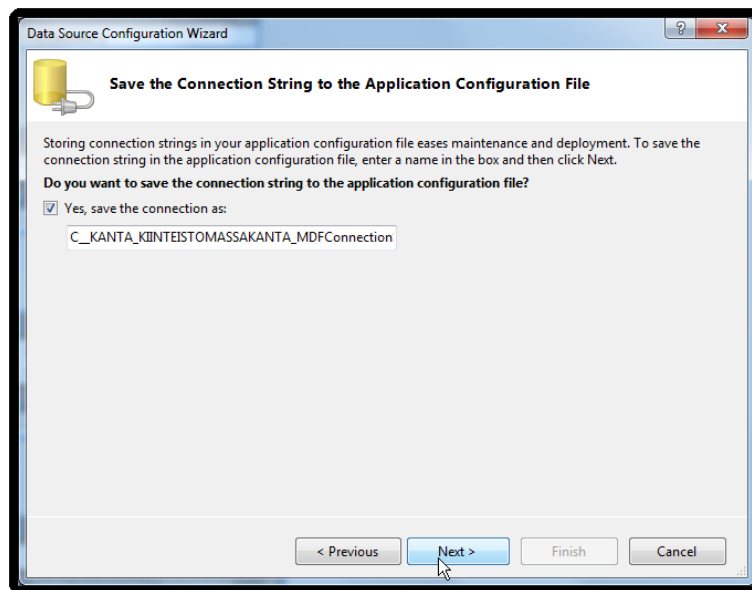
Kuvio 55. Tietokannan tyypin valinta

Näitä näkyy kaksi, koska aiemmin luotu ylimääräinen yhteys näkyy vielä mukana. (Kuvio 56).



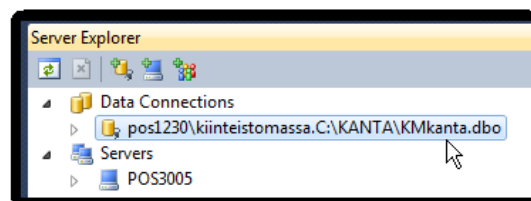
Kuvio 56. Olemassa olevat yhteydet

Tallensin lopuksi yhteydenottorimpsun. Ensimmäisestä kuvasta poiketen vaihdoin nimeksi **KMkanta.dbo** (Kuvio 57).



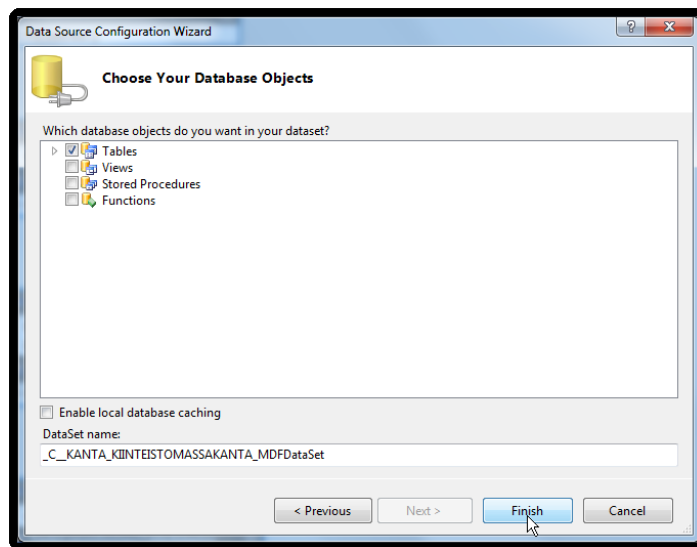
Kuvio 57. Yhteyden tallentaminen

Yhteys näkyi nyt *Visual Studiossa* (Kuvio 58).



Kuvio 58. Yhteys Visual Studiossa

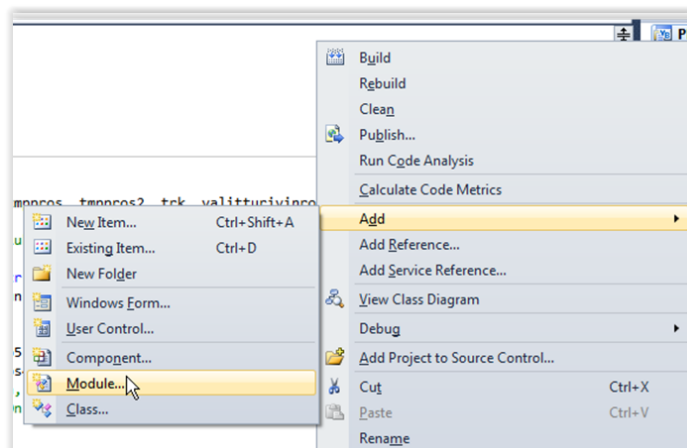
Tämän jälkeen valitsin, mitä halusin kannasta olevan käytettävissä. Pelkät taulut riittivät, joten valitsin vain **Tables**. Välimuistitusta kannasta en tässä vaiheessa halunnut, koska halusin olla varma, että kaikki ohjelmoinnin aikana tekemäni muutokset tulevat/menevät varmasti suoraan kantaan. **Datasetin** nimesin kuvasta poiketen **KMkanta**. Lopuksi klikkasin **Finish** (Kuvio 59).



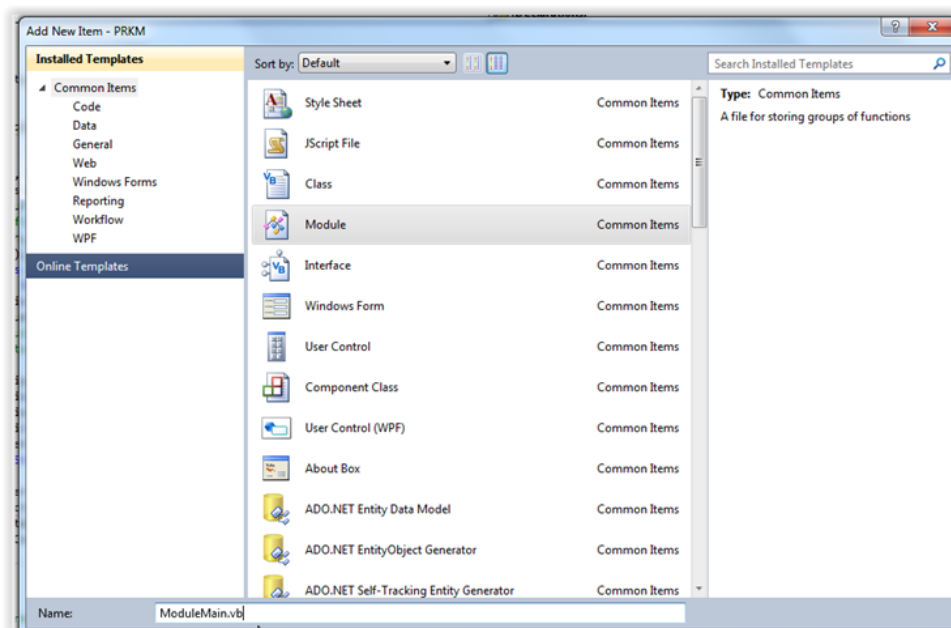
Kuvio 59. Yhteyden konfigurointi

3.3 Moduulin lisääminen projektiin

Tarvitsin moduulin, johon tekisin subeilla yleisimmät komennot, mitä ohjelmassa kaipaisin. Asetin moduulin nimeksi **ModuleMain** (Kuvio 60) ja (Kuvio 61).



Kuvio 60. Moduulin lisääminen projektiin



Kuvio 61. Moduulin lisääminen ja nimeäminen

Moduuleille ei tule omaa formiaan ja niihin luodut subit ovat ajettavissa mistä tahansa formista. Ideana on siis luoda sinne komentoja joita käytetään koko projektissa, jolloin päästään kertakoodauksella. Komennoilla taas säästetään koodia, kun samaa monimutkaista koodisarjaa ei tarvitse toistaa. Kirjoittelin moduuliin heti yleisimmät käyttämäni/luomani komennot.

Olin aiemmin muissa projekteissa luonut monipuoliset tallenna- ja lataa-subit moduuliin, mutta niiden ideana oli lukea ja kirjoittaa tekstitiedostoja. Nyt MS SQL-tietokannan kanssa kyseisiä subeja olisi muokattava, että ne olisivat edelleen käyttökelpoisia.

4 TIETOKANNAN TAULUT

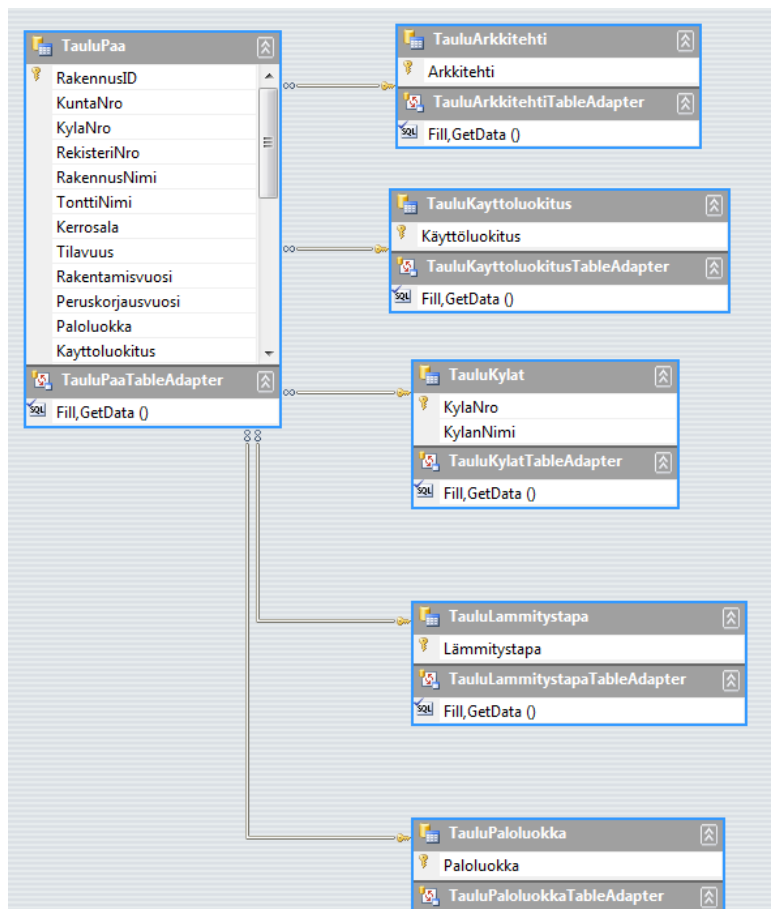
4.1 Alkumäärittelyt ja tarvittavat taulut

Olin pitänyt 10.7.2013 palaverin rakennusmestarien Jorma Varanka ja Ossi Karjalainen kanssa siitä, mitä tietoja kohteista kantaan halutaan. Tällöin kävi ilmi, että sarakkeita päätauluun tarvittiin alustavasti 27 kappaletta. Listaus liitteenä (*Tietokantaa.xls*).

4.2 Luominen ja ominaisuudet

4.2.1 Kannan kuvaaja

Olin luonut projektin *VB.NET*:lla ja tehnyt kuvaajan *KMkanta.xsd*, jota kautta saatoin hallita kannan tauluja ja sarakkeita (Kuvio 62).



Kuvio 62. Kannan kuvaaja

Sama kuvaaja on toki käytettävissä client-projektissakin, mutta ajatuksena näin. Käyttöoikeuksilla voi sitten myöhemmin rajata, mitä sallitaan.

4.2.2 Päätaulun luominen

Loin siis ensimmäiseksi päätaulun (**TauluPaa**), johon ryhdyin luomaan halutut sarakkeet.

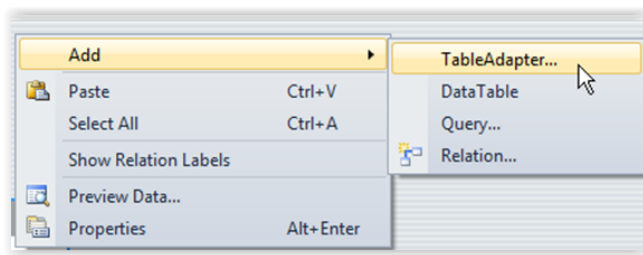
Asetin ensimmäisen sarakkeen nimeksi **RakennusID**-tyypillä **smallint** ja määritin, että arvo tarvitaan. Näitä pakollisia tietoja oli kaikkiaan viisi. Jos datatyyppinä oli merkki, asetin tyypiksi **nvarchar**, koska arvoissa on mm. ääkkösiä. Ennen kuin nimesin sarakkeen, asetin sen hiiren oikean avulla valinnalla ”**Set Primary Key**” -rivin **RakennusID** pääavaimeksi. Kun olin saanut päätaulun valmiiksi, suunnittelin mitä alitauluja tarvitsisin.

Ensimmäisinä mieleen tulivat taulut **paloluokalle, kylälle, käyttöluokitukselle, arkkitehdille ja lämmitystavalle**. Nuo kaikki ovat sellaista tietoa joissa on vain rajallinen määrä vaihtoehtoja, joten niistä voidaan luoda lista (taulu).

Jatkoin lisäämällä loputkin ensin hahmottelemani taulut. Koska useimmiten arkkitehti on jokin jo mainittu, voidaan heille tehdä oma taulunsa.

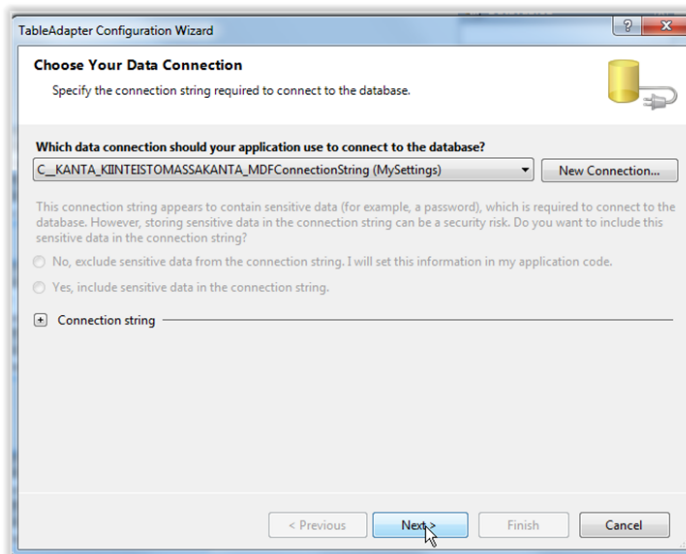
4.3 TableAdapterit

Kun kannassa olivat tarvittavat taulut, niin clientiin loin siihen osoittavan adapterin näin: Otin datasetin päällä hiiren toissijaisella painikkeella ja valitsin ensin **Add TableAdapter** (Kuvio 63).



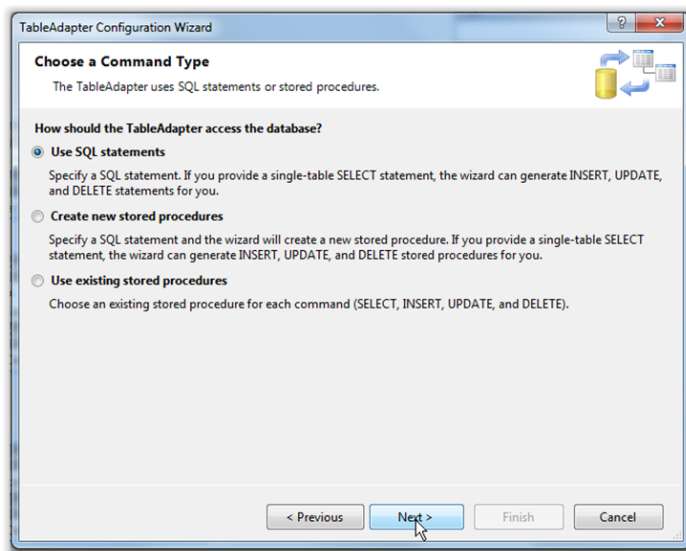
Kuvio 63. TableAdapterin lisääminen

Sitten tarkistin, että kyseessä oli oikea kanta ja klikkasin **Next** (Kuvio 64).



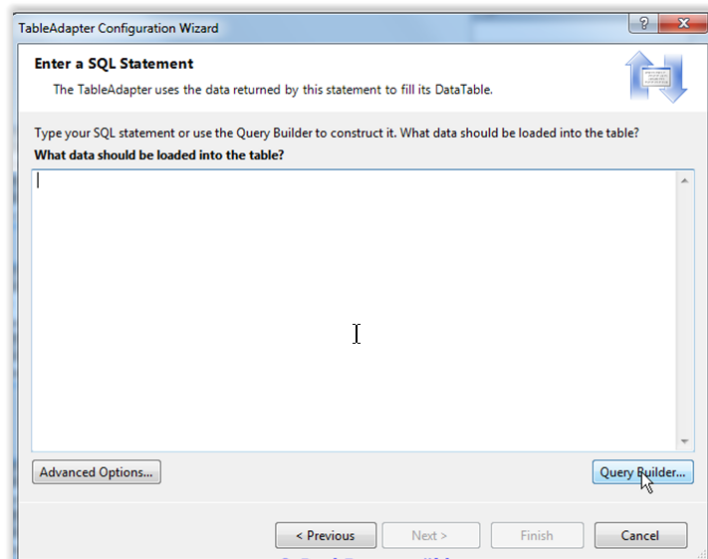
Kuvio 64. TableAdapterin konfigurointi

Next (Kuvio 65).



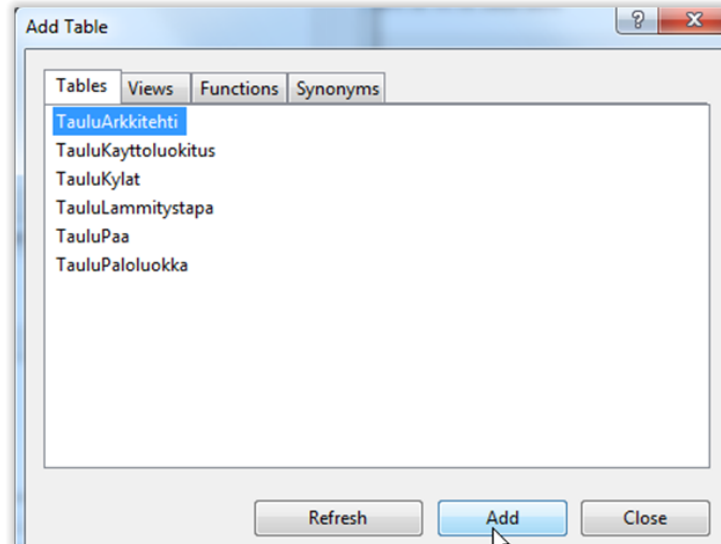
Kuvio 65. TableAdapterin konfigurointi

Tein kyselyn käyttäen ohjattua toimintoa (**Query Builder...**) (Kuvio 66).



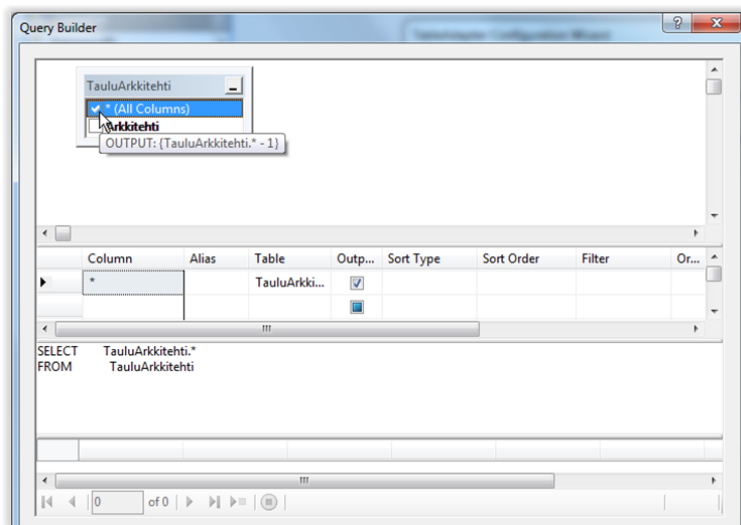
Kuvio 66. TableAdapterin konfigurointi

Valitsin oikean taulun kannasta ja lisäsin sen painikkeella **Add** (Kuvio 67).



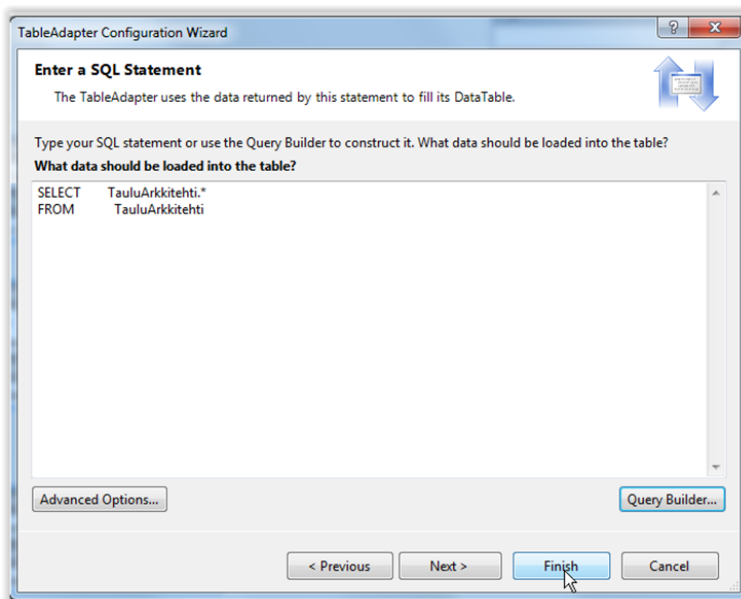
Kuvio 67. Taulun lisääminen

Valitsin, että käytettävissä olivat taulun kaikki sarakkeet (Kuvio 68).



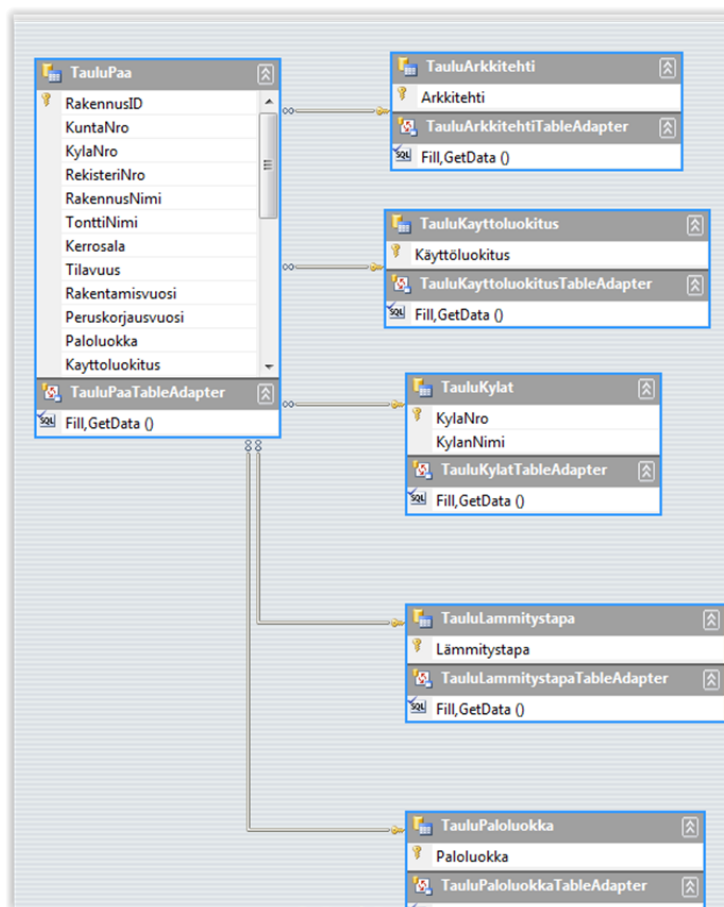
Kuvio 68. Sarakkeiden valinta

Klikkasin **OK** ja lopetin klikkaamalla **Finish** (Kuvio 69).



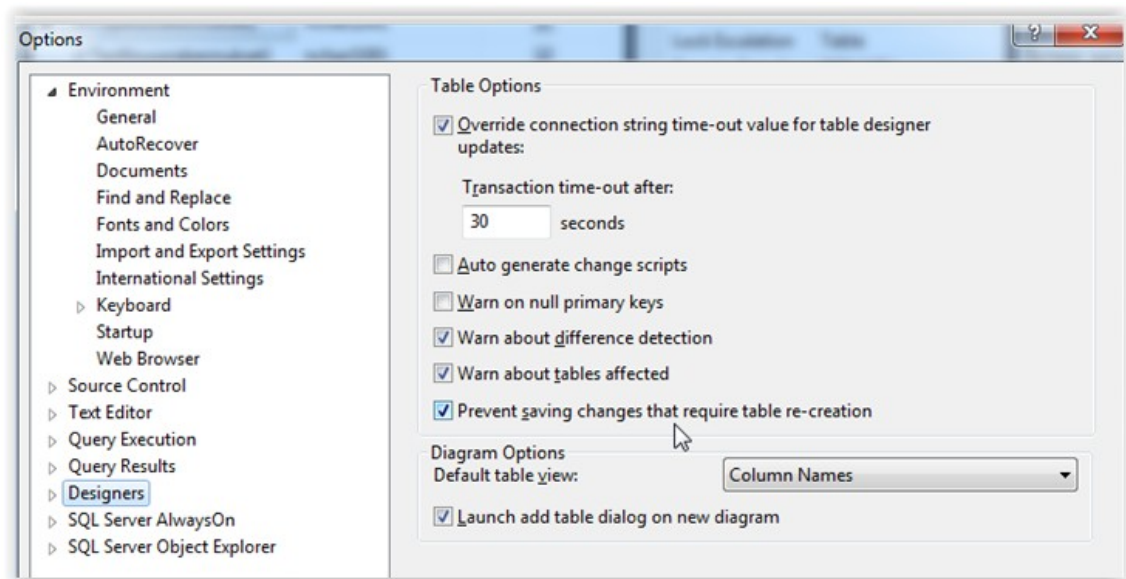
Kuvio 69. Sarakkeiden valinta

Tein saman operaation lopuille tauluille. Nyt näytti siis tältä (Kuvio 70):



Kuvio 70. Kannan kuvaaja

Otin asetuksista pois valinnan kohdasta ”***Prevent saving changes that require table re-creation***”, jotta sain tallennettua juuri tekemäni tietokannan kuvaajan. Tein tämän harjoituksen vuoksi, jotta saan myöhemmin luotua kuvaajan tästä tietokannasta (Kuvio 71).



Kuvio 71. Designerin asetukset

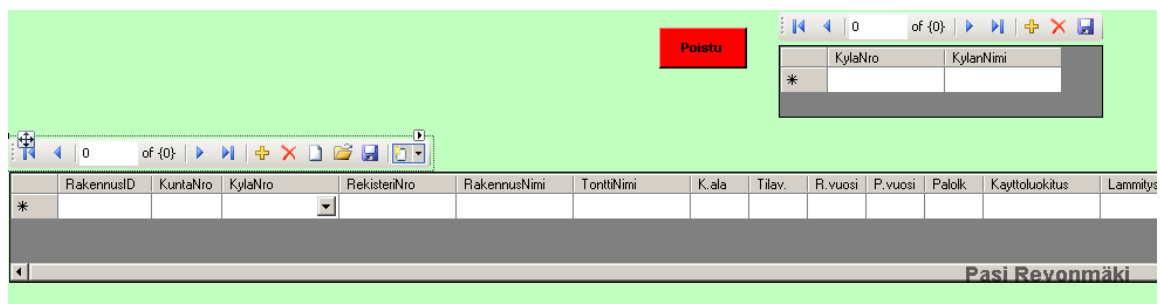
Ohjetta edellisiin löysin täältä:

<http://msdn.microsoft.com/en-us/library/ms233763%28v=vs.90%29.aspx>

5 TIETOJEN VIEMINEN JA TUOMINEN

5.1 Lomakepohjan suunnittelu

Toin pari taulua lomakepohjaan ja aloin hieman hahmotella käyttöliittymää, jolla kantaan viedään tietoja, sekä testaamaan kuinka yhteys kantaan toimii käytännössä. Lisäsin siis formin ja siihen pari **BindingNavigator** -objektia (Kuvio 72).



Kuvio 72. Lomakepohjan valmistelua testausta varten

BindingNavigator -ohjausobjektin konfiguroinnista tarkemmin kappaleessa 5.2.

Ohjetta edellisiin löysin täältä:

<http://msdn.microsoft.com/en-us/library/2b4be09b%28v=vs.90%29.aspx>

BindingNavigatoreihin piti lisätä lähde ja muutama näppäin (kuvin jo tehty tämä toimenpide) lisää (esim. tallennusnäppäin), jotta ne sai ajamaan asiansa.

<http://msdn.microsoft.com/en-us/library/8zhc8d2f%28v=vs.85%29.aspx>

[https://msdn.microsoft.com/en-us/library/safa4957\(v=vs.80\).aspx](https://msdn.microsoft.com/en-us/library/safa4957(v=vs.80).aspx)

Yllä tekemäni toimet piti siis kerrata *MS SQL 2012 SP2 Expressin* kanssa ja tietokantayhteys luoda uudelleen, mutta tässä vaiheessa kaikki toimi uudessakin kokoonpanossa kuten yllä.

Esim. tallennusnapin sain toimimaan tällaisella subilla (Kuvio 73):

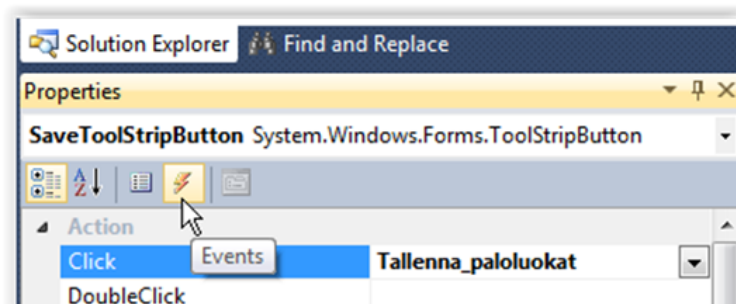
```

Private Sub Tallenna_kylat(ByVal sender As Object, ByVal e As System.EventArgs) Handles ToolStripButton28.Click
    If Me.Validate() = True Then
        Me.TauluKylatBindingSource.EndEdit()
        Me.TauluKylatTableAdapter.Update(Me.KMkanta.TauluKylat)
    Else
        'Mahdollinen virheilmo tähän
    End If
End Sub

```

Kuvio 73. Koodi DataGridViewin tallennusta varten

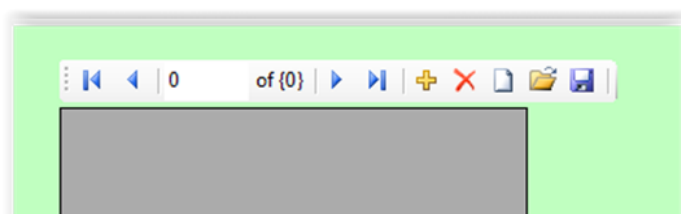
Näppäimen sidoin subeihin eventseissä (kuvassa paloluokan tallennuspainikkeen event) (Kuvio 74).



Kuvio 74. Painikkeen kytkeminen toimintoon

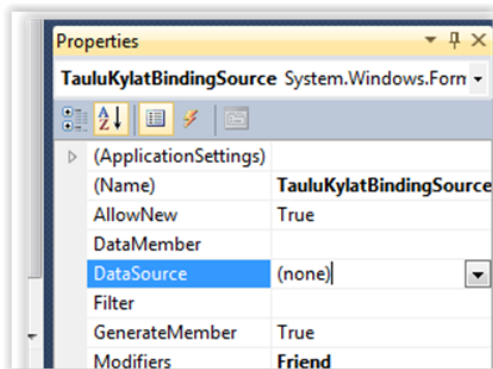
Nyt tietokantapalvelimen alustana oli *Windows 7 64 -bit SP1 FI* ja sama käyttöjärjestelmä myös clientissa, jolla suunnittelin kannan rakenteen ja käyttöliittymän.

Testatessani käyttöliittymää onnistuin myös kadottamaan yhteyden. Tilanne näytti tältä (Kuvio 75):



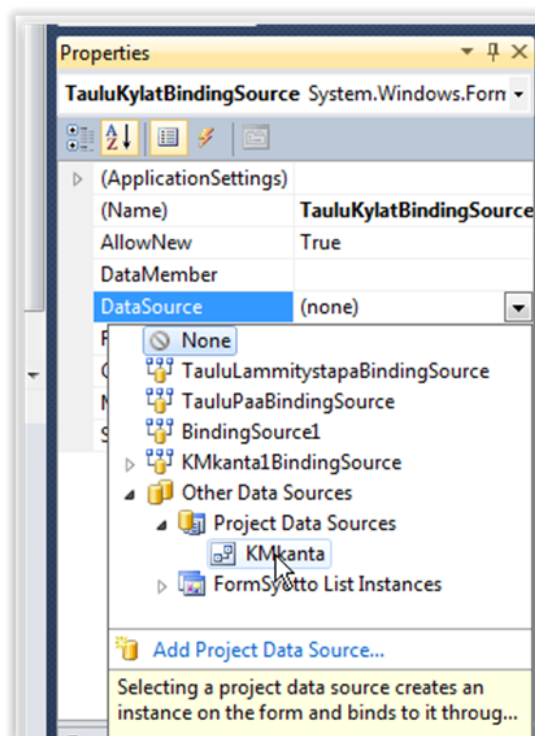
Kuvio 75. Tyhjä DataGridView

Lähdettä ei enää näkynytäkään. Hetken etsittyäni lähdettä huomasin, että **BindingSource**en oma lähde oli kadonnut (Kuvio 76).



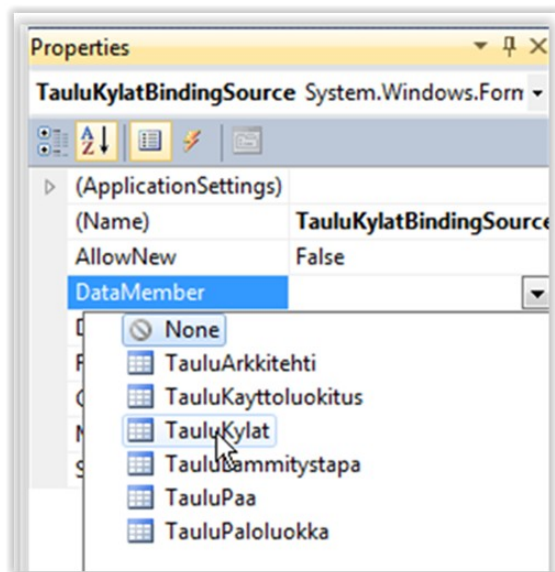
Kuvio 76. DataGridViewin ominaisuudet

Minun piti käydä se hakemassa (alkujaan tätä ei ollut tarvinnut tehdä, vaan se oli tullut automaattisesti) (Kuvio 77).



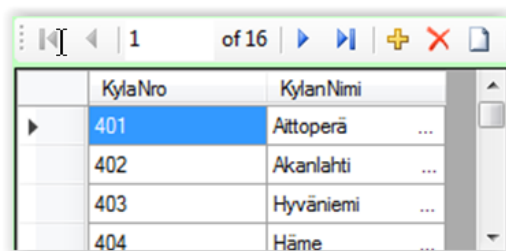
Kuvio 77. Tietolähteen asettaminen DataGridViewille

Samoin **DataMember** piti minun hakea uudelleen (Kuvio 78).



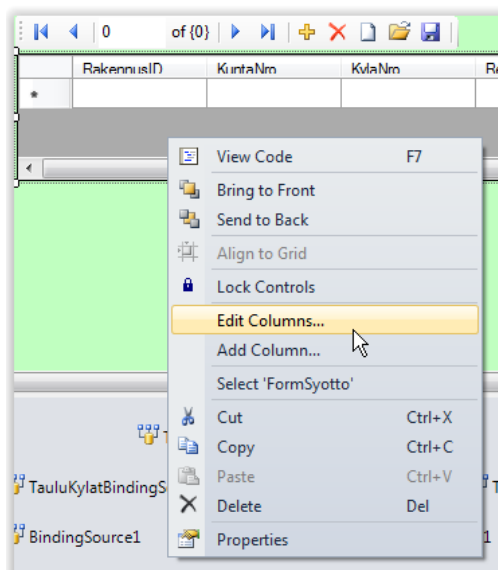
Kuvio 78. Tietolähteen asettaminen

Nyt testissä alkoi kylä-listaus taas näkyä (Kuvio 79).



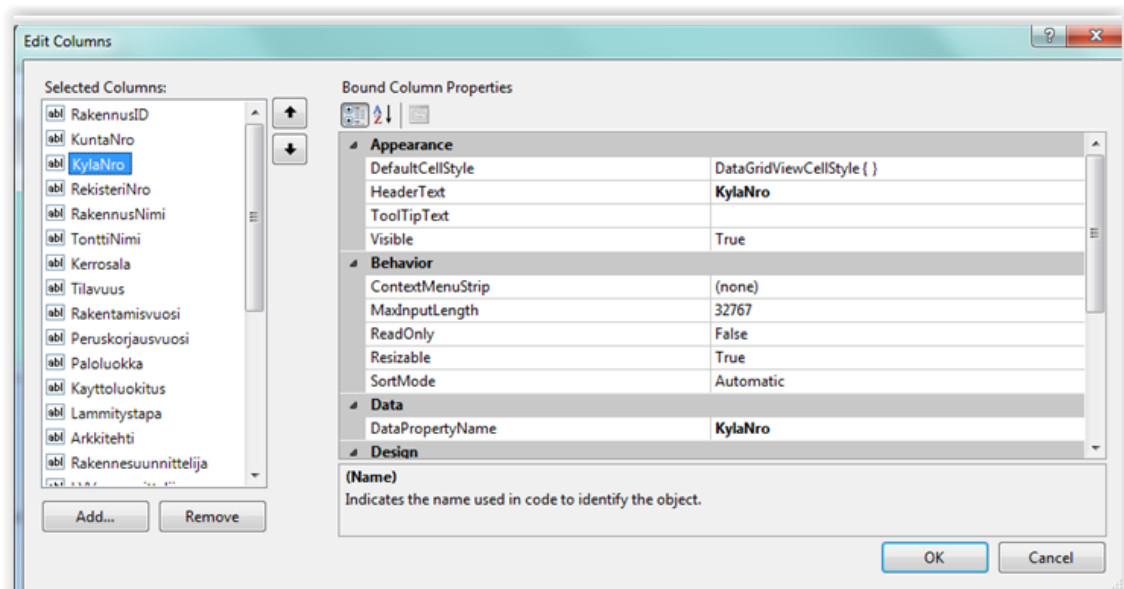
Kuvio 79. DataGridView jossa tiedot kannasta

Kylävalinta oli myös kadonnut **BindingNavigator** -ohjausobjektista. Otin sen **DataGridView**:n ominaisuuksista kyläkohdan tarkasteluun (Kuvio 80).



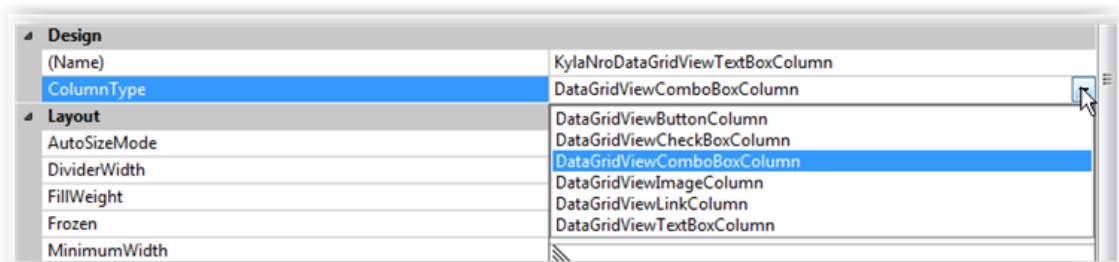
Kuvio 80. DataGridViewin -sarakkeiden konfigurointi

Kyläkohdan sarakkeen säätäminen (Kuvio 81).



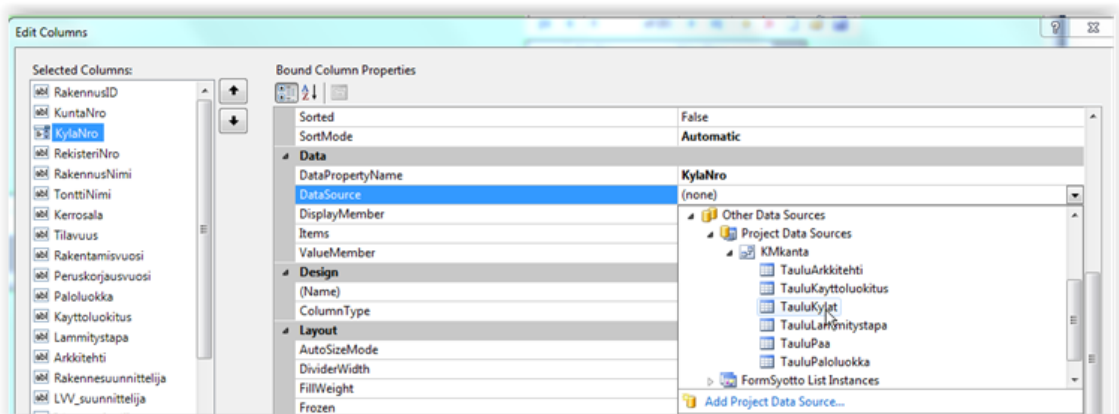
Kuvio 81. Kyläkohdan sarakkeen säätäminen

Asetin tyypiksi **ComboBoxin** (Kuvio 82).



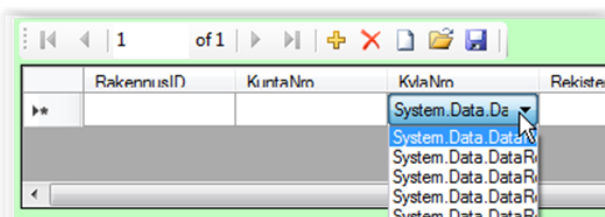
Kuvio 82. Tyypin valinta

Sitten lisäsin lähteeksi taulun, jossa kylät ovat (Kuvio 83).



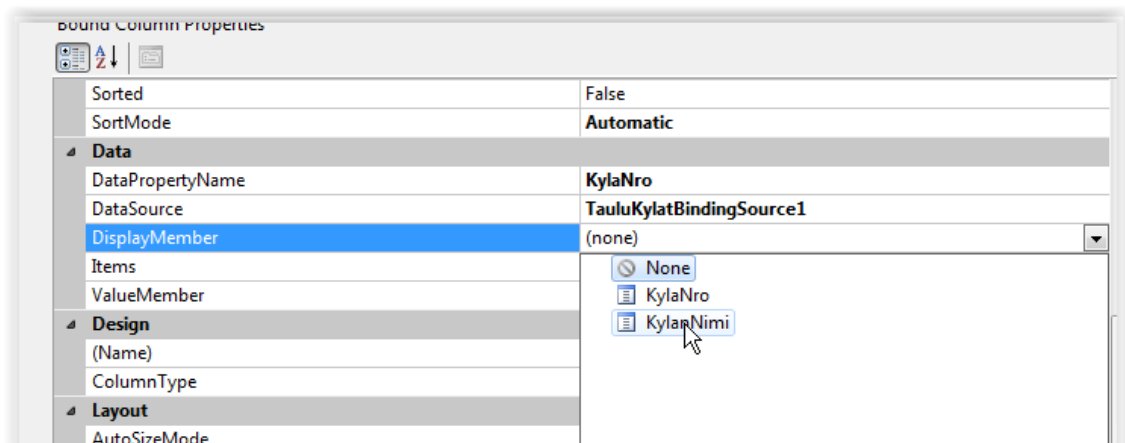
Kuvio 83. Lähteen valinta

Vielä puuttuu jokin arvo (Kuvio 84).



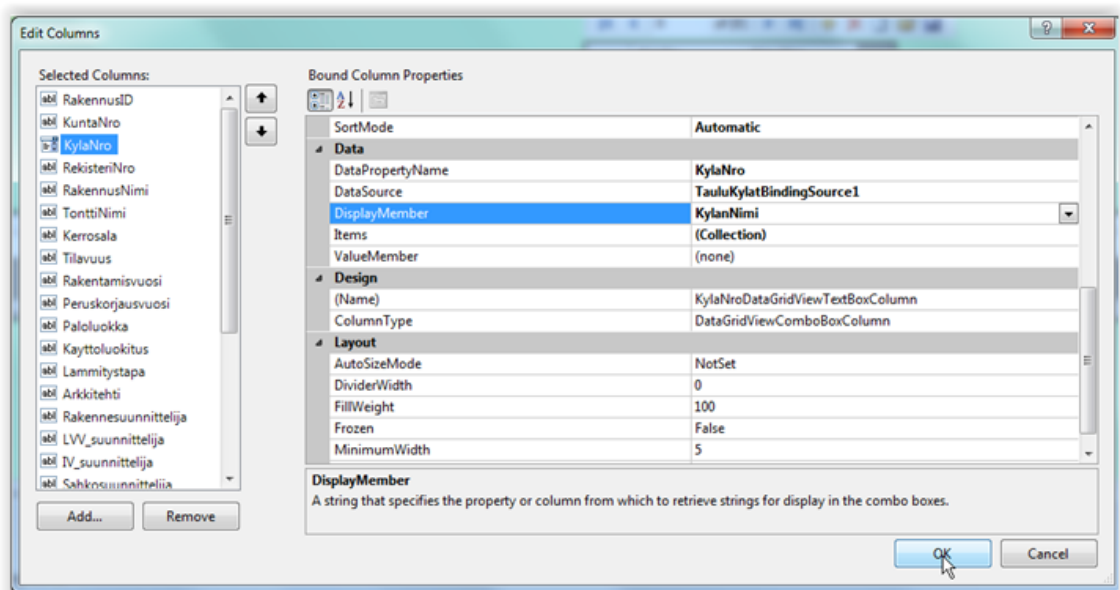
Kuvio 84. Puutteellinen määrittely

DisplayMemberistä puuttui tieto, mikä sarake taulusta näytetään (Kuvio 85).



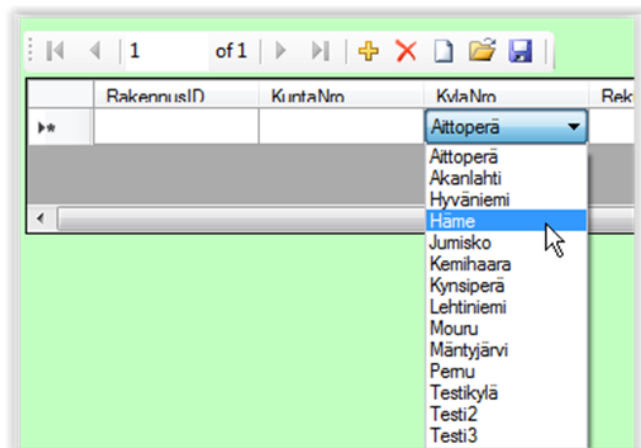
Kuvio 85. DisplayMemberin asettaminen

Asetin sen niin, että näkyy kylän nimellä. Lopputuloksena oli tämä (Kuvio 86):



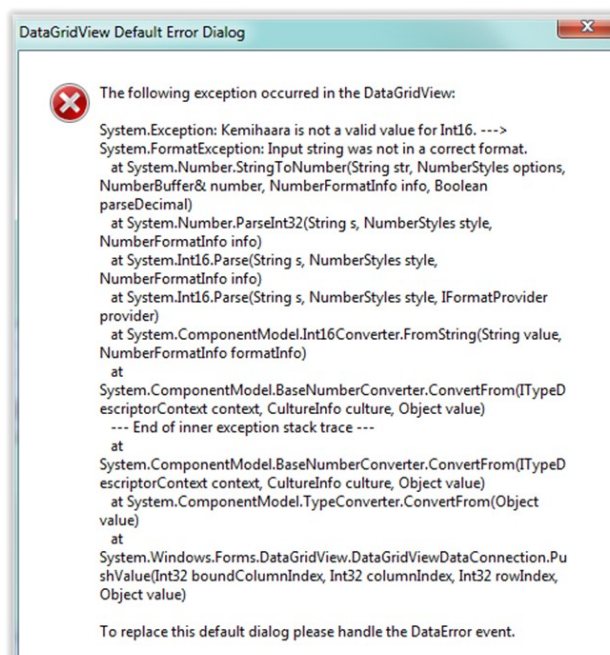
Kuvio 86. Oikeat asetukset sarakkeessa

Käytössä se näyttää tältä (Kuvio 87):



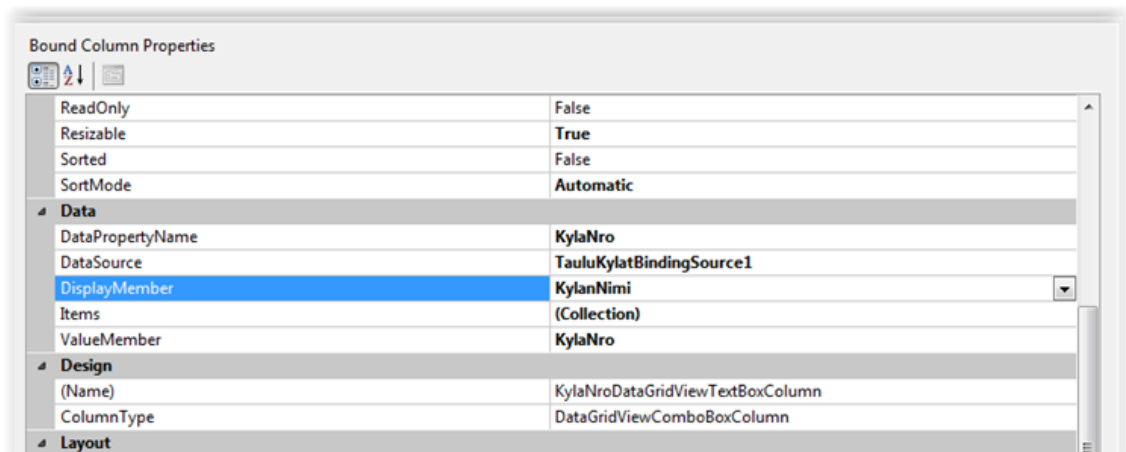
Kuvio 87. Valmis DataGridView

Käynnistin ohjelman ja testasin toimintaa. Heti ilmeni virhe (Kuvio 88).



Kuvio 88. Virheilmoitus DataGridViewista

Vaikutti, että sarakkeen tietotyyppinä oli luku (onhan nimikin **KyläNro**). Pitäisi saada valinta toimimaan niin, että haetaan nimellä, mutta viedään kantaan numerolla. Piti siis säätää vielä **ValueMember** kohdalleen (Kuvio 89).

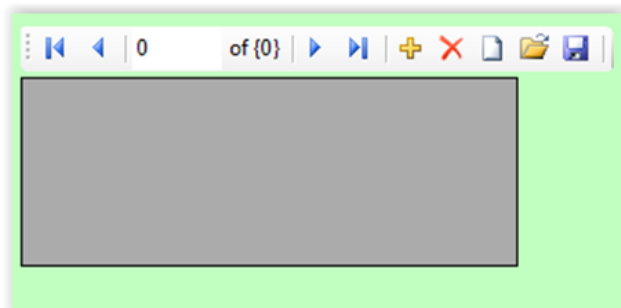


Kuvio 89. ValueMemberin korjaaminen

Nyt ei tule enää siitä virheilmoitusta.

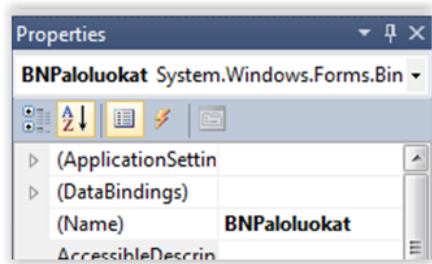
5.2 **BindingNavigator**-ohjausobjektin luominen ja konfigurointi

Seuraavaksi lisäsin uuden **BindingNavigator** -ohjausobjektin paloluokitusta varten. Kaksi edellistähän olivat perua parin vuoden takaisesta tehdystä asennuksesta 2008:lle/XP:lle tehdystä. Ensin siis lisäsin uuden **BindingNavigator**-ohjausobjektin (Kuvio 90).



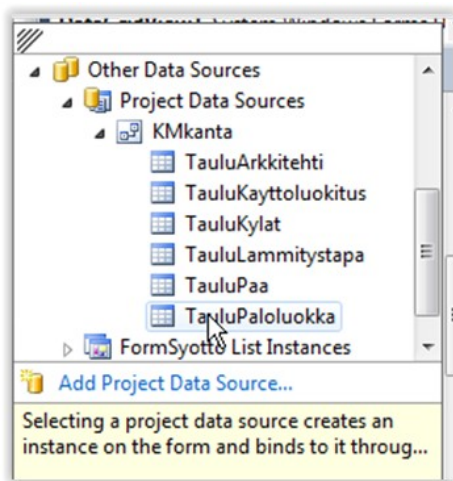
Kuvio 90. Tyhjä BindingNavigator-ohjausobjekti

Sitten nimesin sen uudelleen (Kuvio 91).



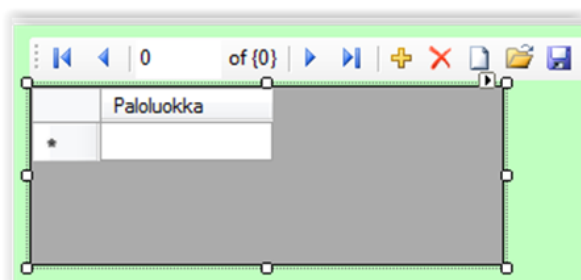
Kuvio 91. Objektin nimi

Määritin lähteen **DataGridView**in kautta (Kuvio 92).



Kuvio 92. Lähteen määrittäminen **DataGridView**ille

Näkymä näytti nyt oikealta (Kuvio 93).



Kuvio 93. Valmis DataGridView

Lisäsin vielä otsikot lomakkeeseen, jotta **BindingNavigator**-ohjausobjektit eivät mene keskenään sekaisin (Kuvio 94).

	KylaNro	KylanNimi
▶	401	Aittoperä ...
	402	Akanlahti ...
	403	Hyväniemi ...
	404	Häme ...

	Paloluokka
*	

Kuvio 94. Otsikot objekteille

6 KÄYTTÖLIITTYMÄ

6.1 Päärakenteesta päättäminen

Alustavasti olin suunnitellut, millaisen käyttöliittymän tiedoille tekisin. Olin miettinyt paria päälinjaa. Joko teen kaikille päätoiminnoille omat forminsa, joiden välillä loikitaan tai sitten kerään ne yhteen lomakkeeseen, koska alustavasti kuusi **BindingNavigator** -ohjausobjektia riittäisi.

Mietin myös sitä mahdollisuutta, että tekisin geneerisen **BindingNavigator**-ohjausobjektin, jonka lähde muuttuisi aina valinnan mukaan. Silloin ehkä selvitäisiin jopa vain kahdella **BindingNavigator**-ohjausobjektilla.

Jos oletuksena on, että samalla ohjelmalla sekä päivitetään tietoja että hyödynetään kantaa, niin käyttöoikeuksien ja muun toiminnan kannalta on järkevintä eriyttää toiminnot omiin lomakkeisiinsa. Se antaa samalla myös skaalautuvuutta. Tämä tuo ohjelmointiin oman ulottuvuutensa, koska nykyinen *VB.NET* ei ole helpoimpia koodattavia multiformina. Onneksi olin jo aiemmin koodannut multiformia, kun olin luonut yritykselleni sen pääohjelmiston.

6.2 **Multiform**-rakenteen luominen

Koska pyörää ei kannata keksiä uudestaan, niin kävin ensin kopioimassa aiemmasta ohjelmastani luokan malliksi (Kuvio 95).

```

Public Class munLomakkeet

    Public Shared FormMain As Form
    Public Shared FormLasku As Form
    Public Shared FormSyotto As Form
    Public Shared FormErittely As Form
    Public Shared FormMatkat As Form
    Public Shared FormRaportti As Form
    Public Shared FormListat As Form
    Public Shared FormEmail As Form
    Public Shared FormMonifirma As Form
    Public Shared FormQ1 As Form
    Public Shared FormQ2 As Form
    Public Shared FormQ3 As Form
    Public Shared FormGoto As Form

End Class

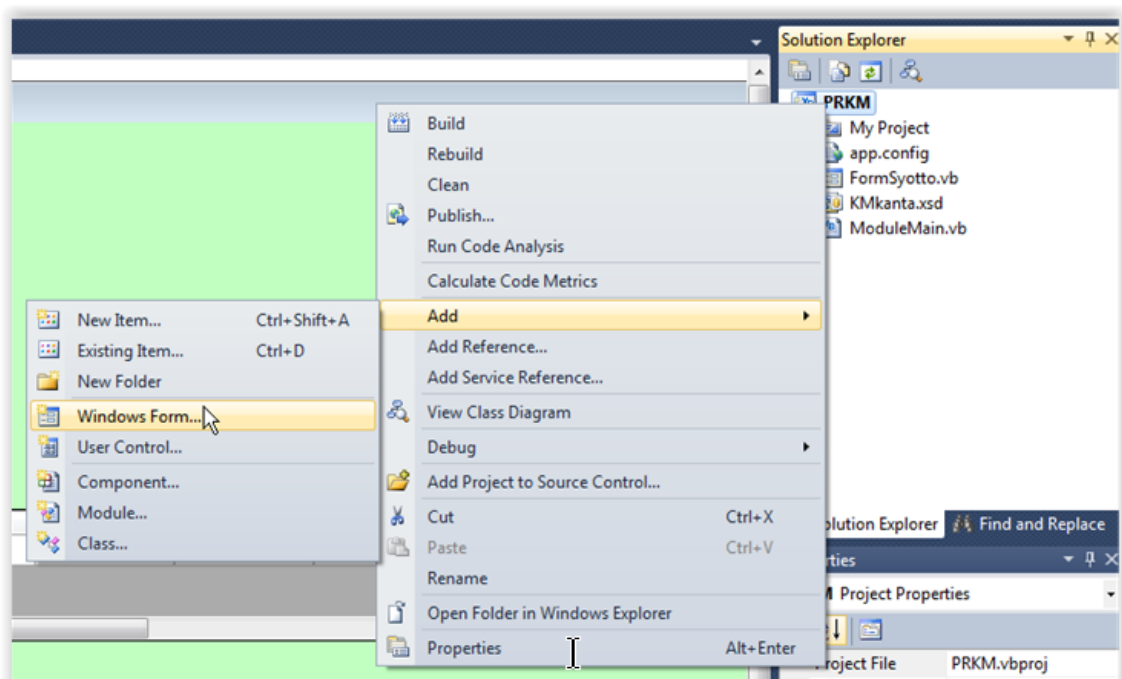
```

Kuvio 95. Lomakeluokka

Lomakeluokkaa pitää tietenkin muokata tähän sopivaksi, mutta se antaa mallin.

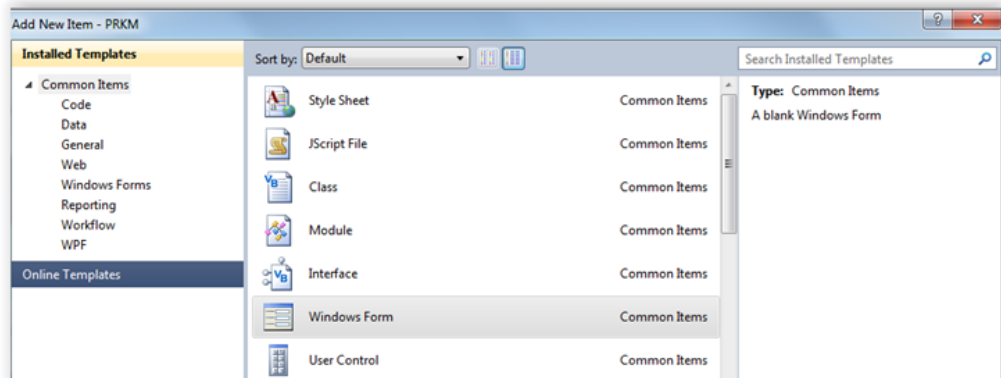
Ensin piti kuitenkin luoda pääformi, jonka koodiin luokan saatoiin sijoittaa. Nykyinen ainoa **FormSyotto** jääköön tietojen syöttämistä varten

Uusi lomake luodaan näin (Kuvio 96):



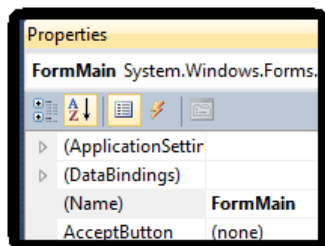
Kuvio 96. Lomakkeen luominen

Klikataan **Add** (Kuvio 97).



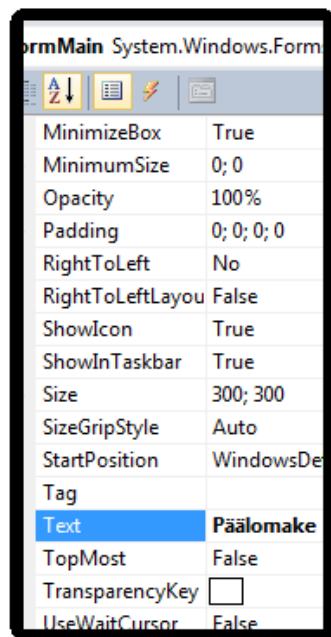
Kuvio 97. Tyypin valitseminen

Nimesin juuri luodun lomakkeen pääformiksi eli **FormMain**. Englanninkielinen sana main on ytimekäs tässä (Kuvio 98):



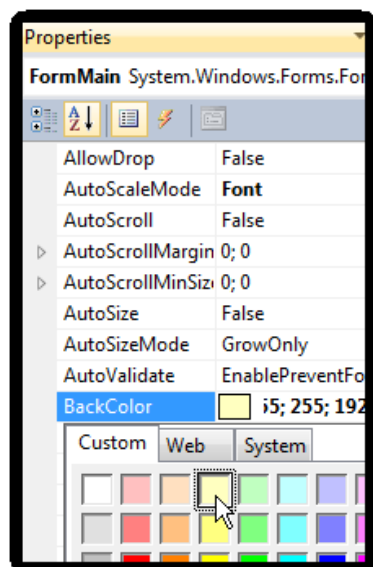
Kuvio 98. Lomakkeen nimi

Lomake otsikoidaan seuraavasti: **Päälomake** (Kuvio 99).



Kuvio 99. Lomakkeen teksti

Vaihdoin taustavärin hieman iloisempaan (Kuvio 100).



Kuvio 100. Lomakkeen taustavärin asettaminen

Menin koodiin tuplaklikkaamalla lomaketta (Kuvio 101).

```

1 Public Class FormMain
2
3     Private Sub FormMain_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load
4     |
5     End Sub
6 End Class

```

Kuvio 101. Lomakkeen latauskoodi

Tässä vaiheessa olin luonut jo ensimmäisen subin, joka ajetaan, kun lomakkeeseen tullaan tai se ladataan.

Nyt mietin alustavasti, että mitä lomakkeita ainakin tarvitsisin. **FormMain** ja **FormSyotto** olivat jo valmiina. Raportin **DataGridViewille** olisi hyvä olla omansa, joten **FormHallinta** ja **FormRaportit** voisivat olla paikallaan.

Lisäsin siis aiemmin kopioimani koodin aiemman perään ja muokkasin tämänhetkisen tilanteen mukaan (Kuvio 102).

```

Public Class munLomakkeet
    Public Shared FormMain As Form
    Public Shared FormSyotto As Form
    Public Shared FormRaportit As Form
    Public Shared FormHallinta As Form

```

Kuvio 102. Uusittu lomakeluokka

Tein lomakkeen **Hallinta** kuten edellä.

Sitten siirsin lomaketiedot jo formin alkuun (Kuvio 103).

```

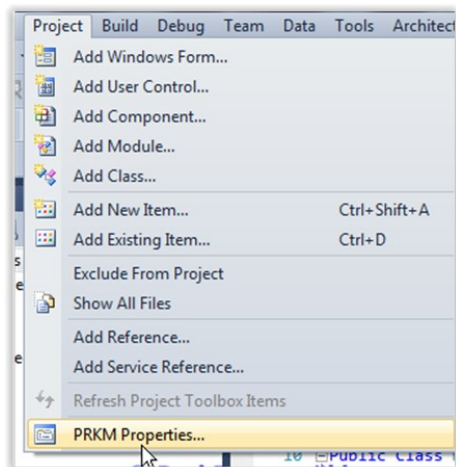
    munLomakkeet.FormHallinta = Me
End Sub

Private Sub ButtonPois_Click(sender As System

```

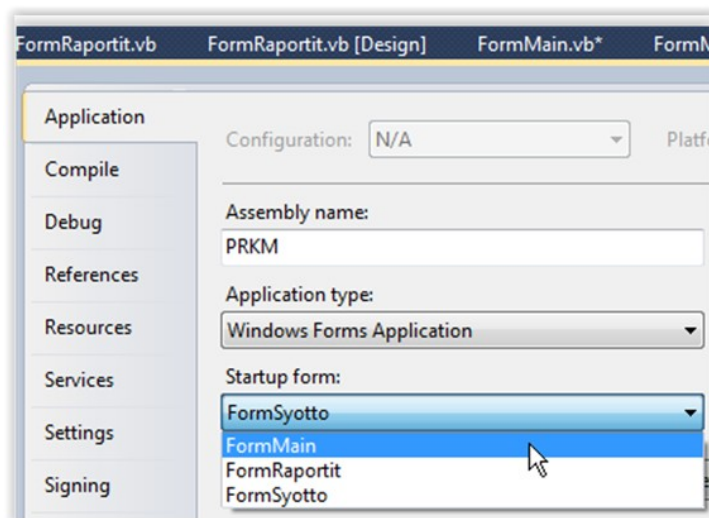
Kuvio 103. Lomakkeen latauskoodi

Määrittelin avautuvaksi lomakkeeksi **FormMainin** (Kuvio 104).



Kuvio 104. Projektin ominaisuudet

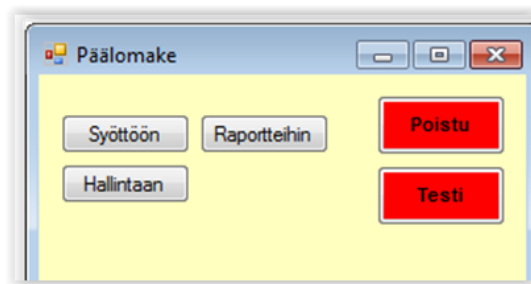
Aloitusslomakkeen asettaminen (Kuvio 105).



Kuvio 105. Aloituslomakkeen asettaminen

Tallensin ja testasin avaamisen. Ohjelma avautui nyt lomakkeeseen **FormMain**, kuten olin tarkoittanutkin.

Seuraavaksi lisäsin pari näppäintä, joilla lomakkeisiin tultaisiin menemään (Kuvio 106).



Kuvio 106. Siirtymänapit

Tuplaklikkasin ko. painiketta ja asetin seuraavan tyyppisen koodin jokaisen painikkeen toimintoon (Kuvio 107):

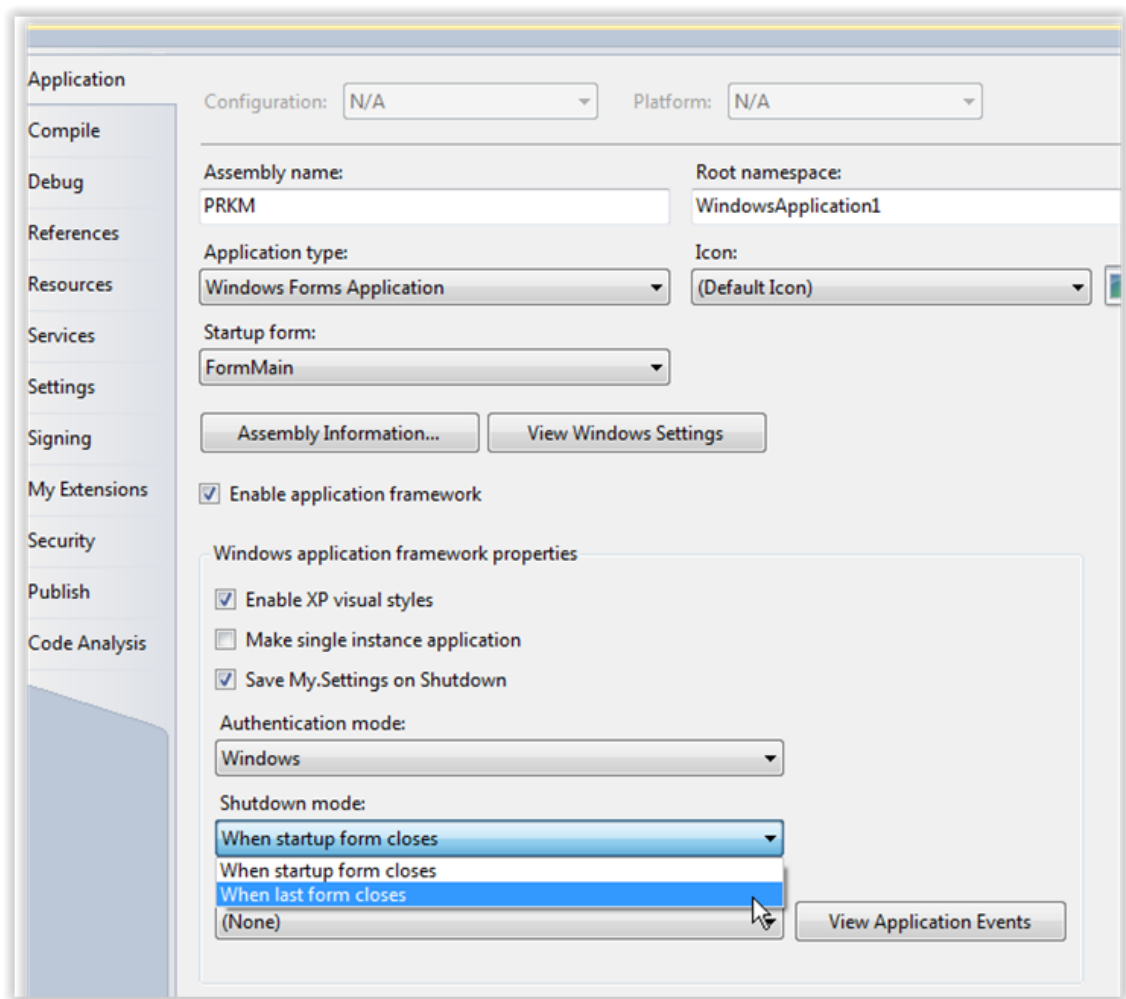
```
Private Sub ButtonSyottoon_Click(sender As System.Object, e As System.EventArgs) Handles ButtonSyottoon.Click
    Dim formUusiFormSyotto As New FormSyotto
    formUusiFormSyotto.Show()
    munLomakkeet.FormMain.Close()

    'Me.Location = New Point(988, 52)
End Sub
' Class
```

Kuvio 107. Siirtymäkoodit

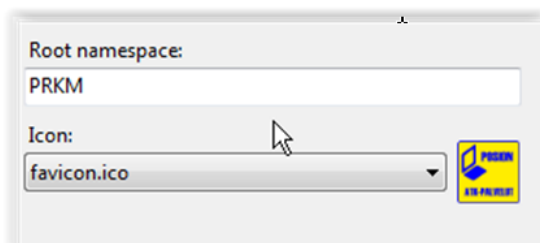
Kuten näkyy, jätin avaussijainnin vielä määrittämättä. Sen ehtii määrittämään myöhemminkin, jos on tarvetta.

Sitten testasin painikkeen. Se ei toimi, koska päälomake sulkeutuu, niin koko ohjelma sulkeutuu. Olin onneksi törmännyt tähän aiemminkin. Projektin ominaisuuksia pitää säätää. Sammutustavaksi pitää muuttaa ”**When last form closes**” eli kun viimeinen lomake suljetaan (Kuvio 108).

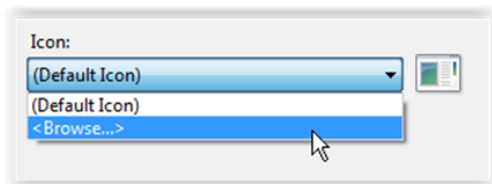


Kuvio 108. Sulkeutumisasetusten määrittäminen

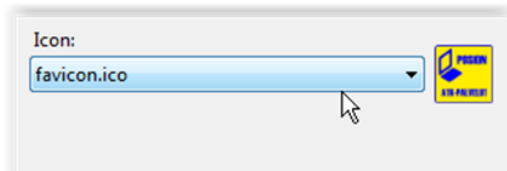
Samalla asetin juurinimen ja ikonin kohdalleen. Ikoniksi tietenkin itse suunnittelemani yrityksen logon (Kuvio 109), (Kuvio 110) ja (Kuvio 111).



Kuvio 109. Juurinimen ja ikonin asettaminen

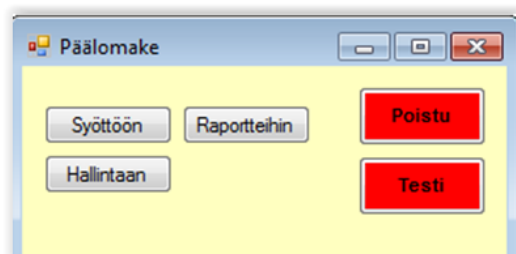


Kuvio 110. Selattavan ikonin hakeminen



Kuvio 111. Ikonin asettaminen

Seuraavaksi tein saman toimenpiteen hallinta- ja raporttilomakkeen painikkeelle. Testasin ja toiminnot onnistuivat. Nyt pääsin mistä tahansa lomakkeesta, mihin tahansa lomakkeeseen ja alkuperäinen lomake sulkeutuu aina (Kuvio 112).

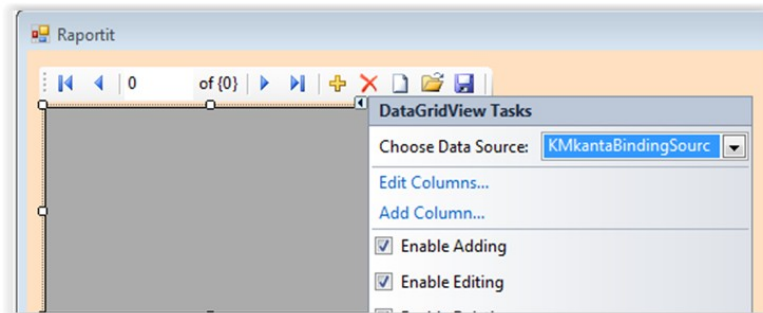


Kuvio 112. Siirtymäpainikkeet

6.3 Tiedot **DataGridView**wissä

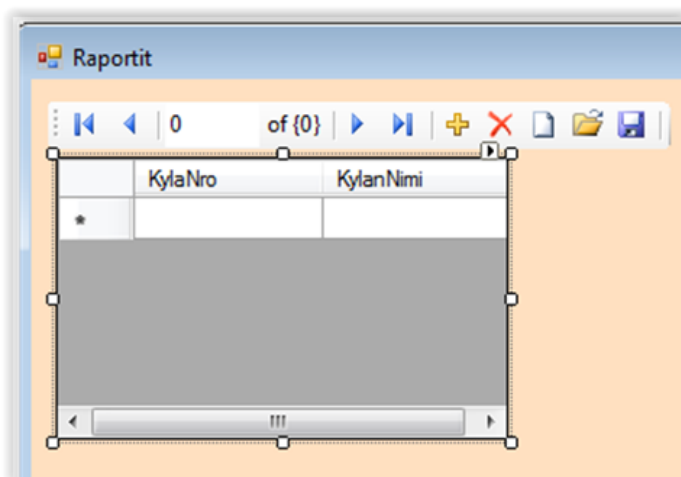
Esimerkiksi hallintalomakkeeseen tuli luoda omat **BindingNavigator**-ohjausobjektit ja **DataGridView**it tietojen näyttämistä varten. Avasin *Hallinta*-lomakkeen ja loin siihen ensin **DataGridView**in ja sen yläpuolelle **BindingNavigator**-ohjausobjektin. Järjestys oli sellainen, että **BindingNavigator**-ohjausobjektille saa valittua lähteen. Se muodostuu automaattisesti, kun **DataGridView**wissä käy valitsemassa lähteen.

Tässä vaiheessa piti siis valita lähde ja näytettävät sarakkeet sekä muokkausmahdollisuudet (kuvakaappaukset raportit -lomakkeesta) (Kuvio 113).



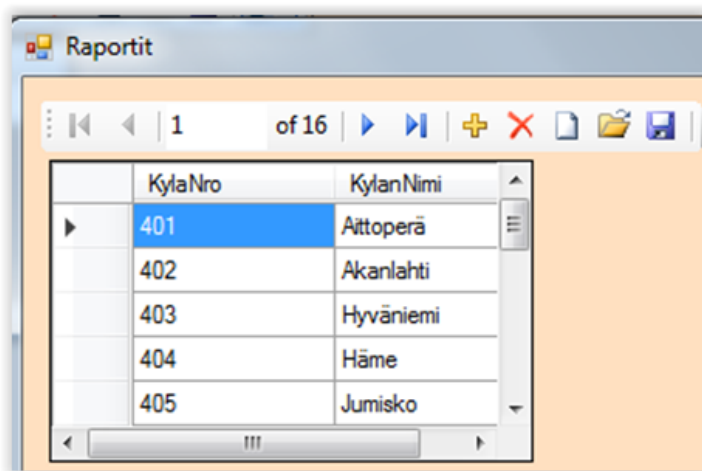
Kuvio 113. Lähteen valinta

Näkymä oli seuraavanlainen (Kuvio 114):



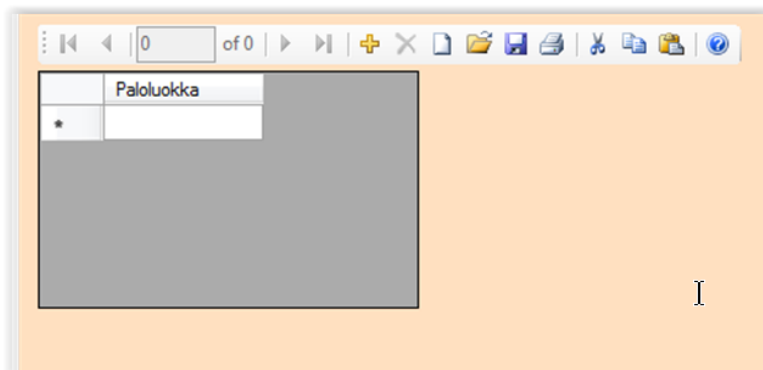
Kuvio 114. Valmis DataGridView

Käynnistettynä näkymä oli tämä (Kuvio 115):



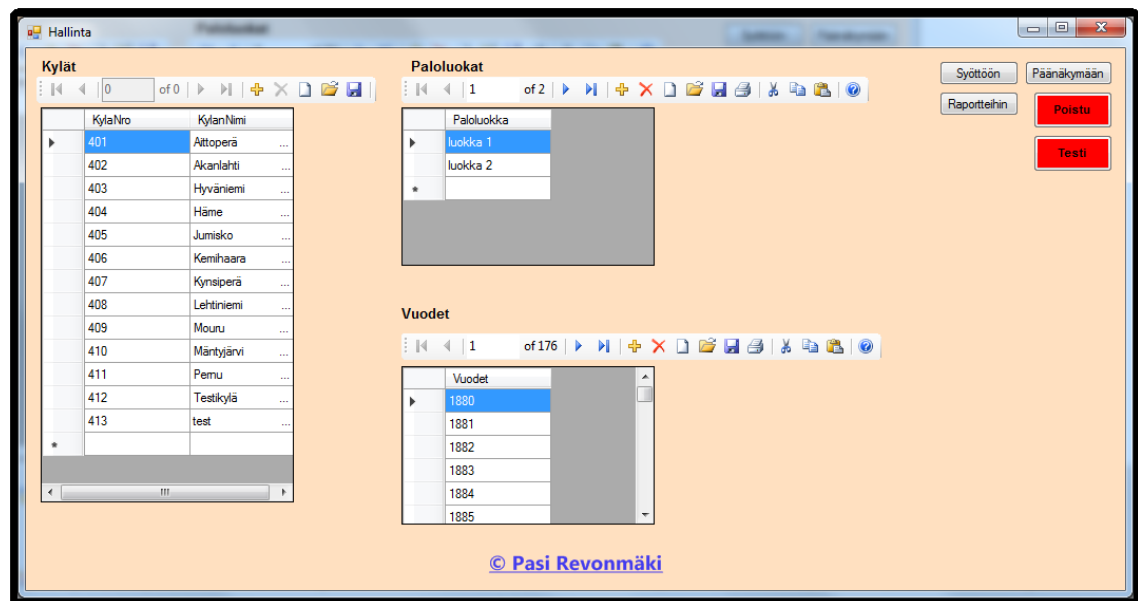
Kuvio 115. Kylät DataGridView

Sitten suoritin saman toimenpiteen paloluokille (laitoin näkyviin kaikki vakioikonit hiiren toissijaisella näppäimellä) (Kuvio 116).



Kuvio 116. Paloluokka DataGridView

Lisäsin pari luokkaa, tallensin, suljin ja avasin. Toimenpide onnitui (Kuvio 117).



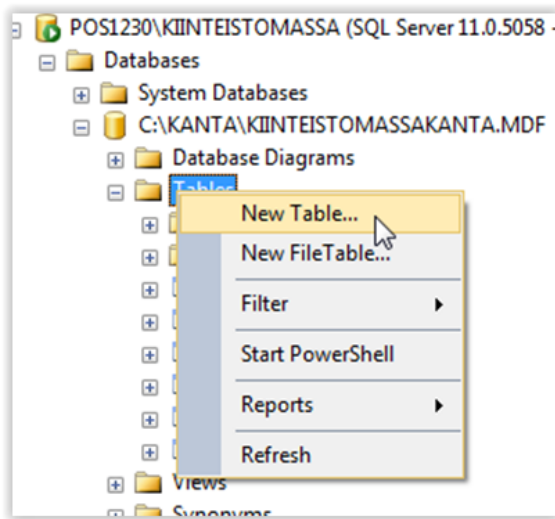
Kuvio 117. Valmis lomake

Kuten kuviosta huomataan, niin lisäsin jo myös painikkeet **Poistu** ja **Testi** joka lomakkeeseen. Poistu-näppäimeen tein subin, joten näppäin myös toimii ja sillä saa sovelluksen suljettua.

6.4 Alitaulujen hallintalomakkeen käyttöliittymän suunnittelu

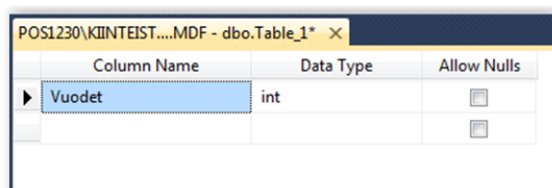
Kuten aiemmassa kohdassa jo nähtiin, alkoi käyttöliittymä muotoutua. Seuraavaksi lisäsin samalla tavalla **BindingNavigator**-ohjausobjektit ja **DataGridView**it käyttöluokitukselle, lämmitystavalle ja arkkitehdeille. Käyttäjät halusivat valita myös vuoden valmiista valikosta, joten sille tulisi tehdä lisätaulu kantaan.

Avasin siis palvelimessa *Microsoft SQL Server Management Studio 2012*:n ja lisäsin taulun (Kuvio 118).



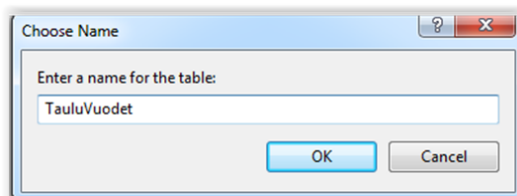
Kuvio 118. Taulun lisääminen

Asetin sarakkeen nimeksi **Vuodet**, taulun nimeksi **TauluVuodet** ja tyyppiä **int**, sekä estin nollan käytön (Kuvio 119).



Kuvio 119. Sarakkeen lisääminen

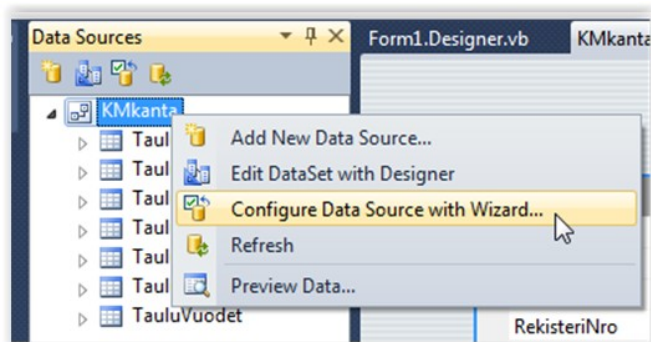
Nimesin taulun (Kuvio 120).



Kuvio 120. Taulun nimeäminen

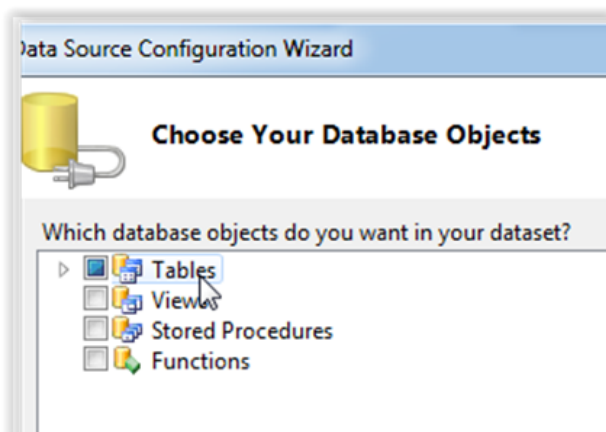
Luomani taulu olisi sitten käytettävissä projektissani. **Datasource** vain piti tässä välissä poistaa ja lisätä uudestaan, jotta juuri lisätyn taulun sai mukaan. Tämän tein tietenkin molemmille projekteille sekä serverillä olevalle että clientille. Tämä

voidaan tehdä vaihtoehtoisesti helpommin menemällä niissä läpi konfigurointivelho (Kuvio 121).

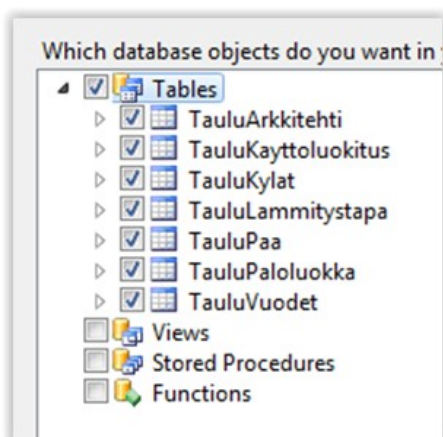


Kuvio 121. Tietolähteen määrittely

Kävin valitsemassa kaikki taulukot (Kuvio 122) ja (Kuvio 123).



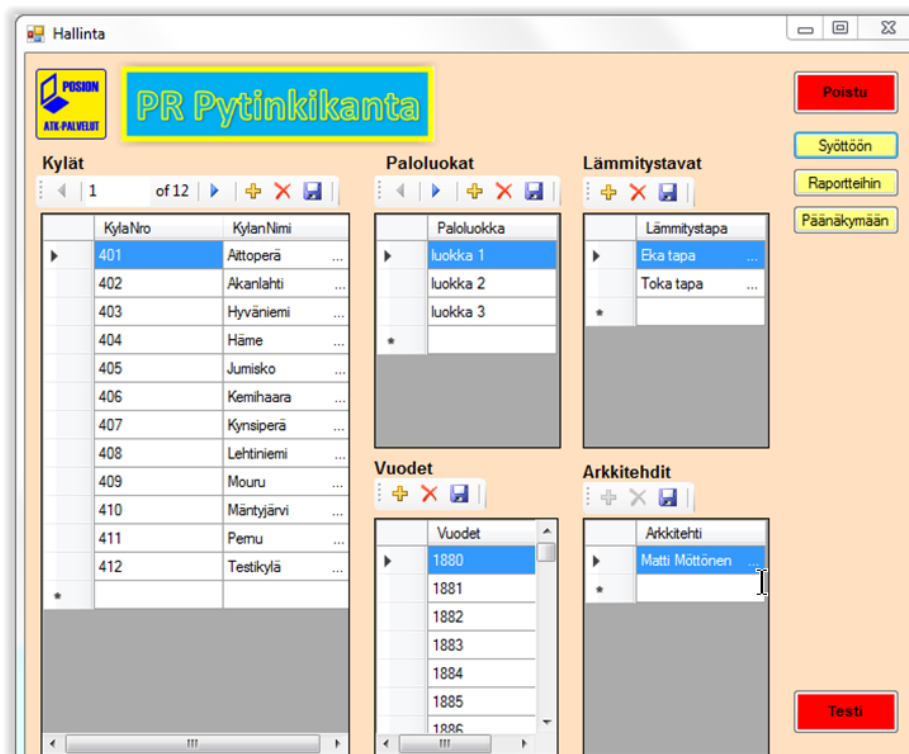
Kuvio 122. Taulujen valinta



Kuvio 123. Taulujen valinta

Näin sain uuden taulun **TauluVuodet** myös käyttöön. Lisäsin serveriprojektissa **BindingNavigator**-ohjausobjektin ja **DataGridView**in vuosille sekä koodin painikkeelle, jolla muutokset viedään kantaan.

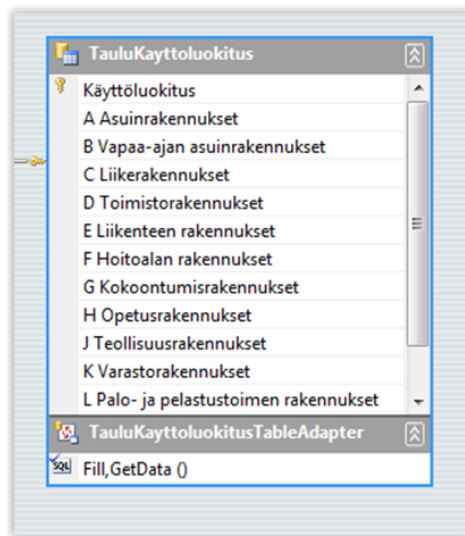
Tein loputkin **BindingNavigator**-ohjausobjektit ja **DataGridView**it, jotta sain tehtyä päivitysliittymän jokaiselle alitaululle. Lisäsin lämmitystavat ja arkkitehdit. Samalla siistin muutakin ja lisäsin firman linkin ja logot (Kuvio 124).



Kuvio 124. Valmis hallintalomake

Tässä vaiheessa rakennusmestari kävi kertomassa, että tarvitaan vielä taulu käyttöluokitukselle ja sen lisäksi vielä useita sarakkeita käyttöluokitusten luokituksille. Sain mallin aiemmasta **vba**:lla tekemästani ohjelmasta, jota käytettiin **Excel 2003**:lla.

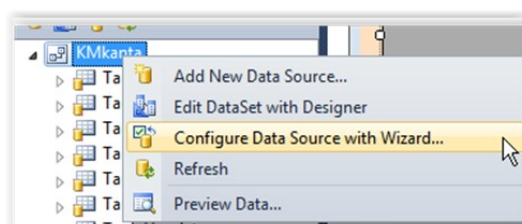
Lisäsin **Management Studiolla** taulun. Kuvaajasta tuli tällainen (Kuvio 125):



Kuvio 125. Käyttöluokitustaulun kuvaaja

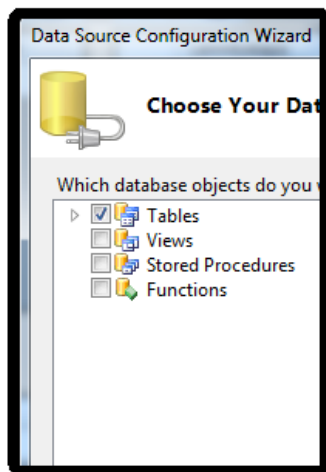
Sitten aloin muokata taas *Hallinta*-lomaketta. Tässä vaiheessa opin, ettei uutta **DataGridView**iä kannata lähteä tekemään kylmiltään, koska tällöin tulee uusi tietolähde ja vanhat **DataGridView**it sekoavat helposti. Muutenkin tulee ylimääräisiä kontrollereita lomakkeeseen suunnittelupuolelle.

Kun uusi taulu oli lisätty niin lähdinkin päivittämään **Data Sourcea** ja kävin siellä lisäämässä taas kaikki taulut käytettäväksi (Kuvio 126).



Kuvio 126. Tietolähteen määrittely

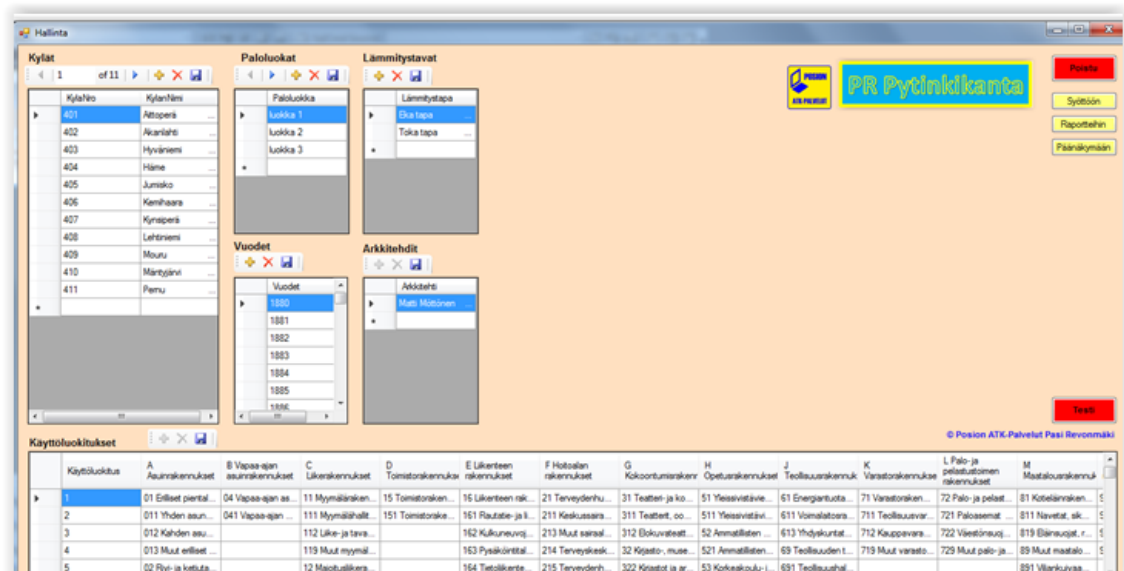
Taulujen valinta (Kuvio 127).



Kuvio 127. Taulujen valinta

Lisäsin uuden **DataGridViewin** ja kävin suoraan koodissa muuttamassa, että lähde on **KMkanta** eikä ohjelman juuri turhaan luoma **KMkanta1**. Sitten poistin ylimääräisen lähteen **KMkanta1**. Korjailin tallennuksen koodin kuntoon ja syötin tauluun kaikki ne luokitukset, mitä *Excel*issäkin oli.

Lopultakin alkoi näyttää siltä miltä pitää (Kuvio 128).



Kuvio 128. Valmis hallintalomake

Tätä lomaketta tullaan käyttämään vähän, koska sillä vain hallinnoidaan alitauluja eli lisätään/päivitetään/poistetaan tietoja, jotka eivät juurikaan tule muuttu-

maan. suunnittelin myös alustavasti, asetanko lomakkeen salasanan taakse tai puhtaasti serverillä sijaitsevaan clienttiin. Salasanalomakkeen ja sen koodauksen olen luonut jo aiemmissa projekteissani, joten sen lisääminen ei olisi iso ponnistus.

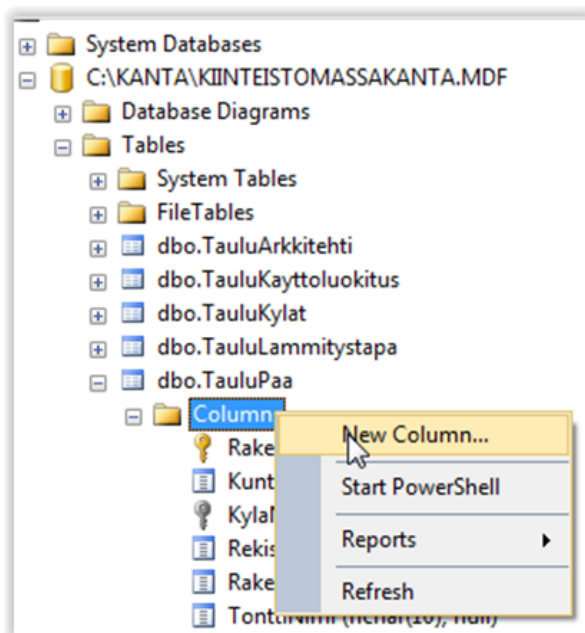
6.5 Päätaulun syöttölomakkeen käyttöliittymän suunnittelu

Koska yksittäisestä kohteesta kirjataan 28 tietoa, ei yksi ***BindingNavigator***-ohjausobjekti ole järkevin tapa tietojen viemiseen. Hahmottelin, että teen syöttämislle pystylomakkeen, johon jokaiselle tiedolle on oma objektinsa. Samoin kyseiseen lomakkeeseen ei ole järkevää laittaa muiden alitaulujen tietojen päivitystä, koska päätaulu itsessään vie runsaasti tilaa.

Lisäsin yhden lomakkeen, loin sen lomakkeen siirtymisille painikkeet ja päivitin luokkakoodin. Samalla keksin, että voisin luoda kaksi saraketta päätauluun lisää. Toiseen sarakkeeseen laitettaisiin tarvittaessa linkki siihen kansioon, missä kohteesta skannatut asiakirjat tms. sijaitsisivat. Kansio tultaisiin kopioimaan/siirtämään tietokantapalvelimella sijaitsevaan kanta-kansioon.

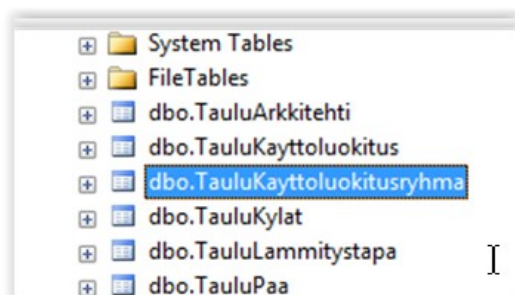
Näin siksi, että myöhemmin tulevat muutokset *AD*:ssa eivät vaikuttaisi linkityksiin ja rikkoisi niitä. Sovin tästä teknisen toimen henkilöstön kanssa. Toisen sarakkeen ajattelin tehdä huomautuksia varten vapaalle tekstille.

Olen oppinut, että sellainen on poikkeuksetta tarpeen. Nämä hoituivat *Management Studiolla* (Kuvio 129).



Kuvio 129. Sarakkeen lisääminen

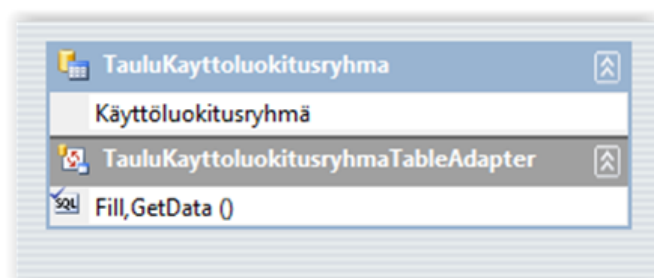
Kuten aiemmin totesin *Hallinta*-lomakkeen kohdalla, niin nyt paljastuneet käyttöluokitusvaatimukset teettivät lisätyötä. Päätaulun syöttölomakkeeseen sen vaikutus oli, että tarvitaan **ComboFix**-objekti, jossa valintoina olivat käyttöluokituksen pääryhmät. Tästä taas seuraa vaatimus, että niille on tehtävä oma taulukkonsa ja päätauluun on lisättävä yksi sarake ja kuvaajat niille. Joten tein ensin ne *Management Studiolla* aiemman esimerkin mukaisesti. Samoin lisäsin sarakkeen **Muut** vapaalle tekstille (Kuvio 130), (Kuvio 131) ja (Kuvio 132).



Kuvio 130. Sarakkeen lisääminen

	Column Name	Data Type	Allow Nulls
🔑	RakennusID	smallint	<input type="checkbox"/>
	KuntaNro	smallint	<input type="checkbox"/>
	KylaNro	smallint	<input type="checkbox"/>
	RekisteriNro	smallint	<input type="checkbox"/>
	RakennusNimi	nvarchar(50)	<input type="checkbox"/>
	TonttiNimi	nvarchar(50)	<input checked="" type="checkbox"/>
	Kerrosala	smallint	<input checked="" type="checkbox"/>
	Tilavuus	smallint	<input checked="" type="checkbox"/>
	Rakentamisvuosi	smallint	<input checked="" type="checkbox"/>
	Peruskorjausvuosi	smallint	<input checked="" type="checkbox"/>
	Paloluokka	nvarchar(50)	<input checked="" type="checkbox"/>
▶	Kayttoluokitusryhma	nvarchar(50)	<input checked="" type="checkbox"/>
	Kayttoluokitus	nvarchar(50)	<input checked="" type="checkbox"/>
	Lammitystapa	nvarchar(50)	<input checked="" type="checkbox"/>
	Arkkitehti	nvarchar(50)	<input checked="" type="checkbox"/>

Kuvio 131. Sarakkeen lisääminen



Kuvio 132. Uuden sarakkeen kuvaaja

Myöhemmin muutin osan datatyypeistä muotoon **nvarchar(MAX)**.

Tein syöttölomakkeen alustavasti valmiiksi. Lisäilin joka kohdalle oman syöttöobjektin ja kytkin ne tarpeen mukaan tietokannan tauluihin, jolloin sain kannasta valmiit listat. Kaikkiin **ComboBox**-objekteihin kerätään tiedot Valinnat-taulusta. Tilanne näytti tässä vaiheessa tältä (Kuvio 133):

The screenshot shows a web application window titled 'Tietojen syöttö' (Data Entry). The main header includes the 'PR Pytinkikanta' logo and three navigation buttons: 'Päänäkymään' (Main View), 'Raportteihin' (Reports), and 'Hallintaan' (Management). A red 'Poistu' (Logout) button is in the top right corner, with a 'Testi' button below it.

The form contains two columns of input fields:

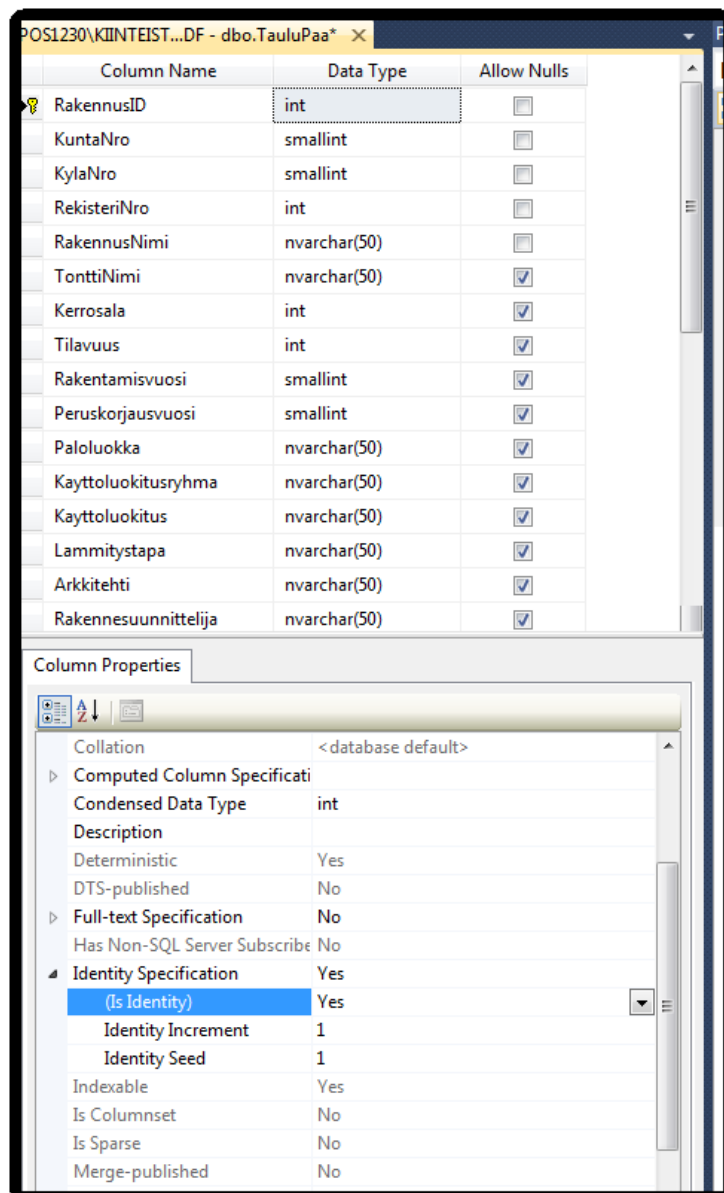
- Left Column:** ID (text), Kuntanro (614), Kylä (dropdown), Rekisterinro (text), Rakennuksen nimi (text), Tontin nimi (text), Kerrosala (text), Tilavuus (text), Rakentamivuosi (dropdown), Peruskorjausvuosi (dropdown), Paloluokka (dropdown), Käyttöluokitusryhmä (dropdown menu open), Käyttöluokitus (text), Lammitystapa (text), and Arkkitehti (text).
- Right Column:** Rakennesuunnittelija (text), LVV-suunnittelija (text), IV-suunnittelija (text), Sähkösuunnittelija (text), Automaatiosuunnittelija (text), Erikoissuunnittelija (text), Rakennusurakoitsija (text), LVI-urakoitsija (text), LVV-urakoitsija (text), IV-urakoitsija (text), Sähköurakoitsija (text), Automaatiourakoitsija (text), Erikoisurakoitsija (text), Kiinteistönhoito (text), and Linkitys (text).

The 'Käyttöluokitusryhmä' dropdown menu is open, showing a list of building types: A Asuinrakennukset, B Vapaa-ajan asuinrakennukset, C Liikerakennukset, D Toimistorakennukset, E Liikenteen rakennukset, F Hoitoalan rakennukset, G Kokoontumisrakennukset, H Opetusrakennukset, and I Teollisuusrakennukset.

At the bottom, there is a copyright notice: '© Pasi Revonmäki' and '© Posion ATK-Palvelut Pasi Revonmäki'.

Kuvio 133. Valmis syöttölomake

Halusin ID:n automaattisesti juoksevaksi, mistä linkissä on ohje: <http://stackoverflow.com/questions/10991894/auto-increment-primary-key-in-sql-server-management-studio-2012> (Kuvio 134).



Kuvio 134. Juokseva numerointi

Vielä puuttui tallenna-näppäin ja sen koodi. Siitä tulikin hieman aiempaa monimutkaisempi, koska nythän ei ollut kyse yhden objektin sisältämien tietojen tallennuksesta, vaan kokonaisen lomakkeen sisältämien tietojen keräämisestä ja tallentamisesta taulukkoon. Muutin vuosien tietotyyppiä **nvarchar(4)**, jotta sain helposti hyväksytettyä tyhjän arvon. Datatyyppi **int** ei sallisi tyhjää. Sama toimenpide tehdään kylänumerolle. Sen lisäksi tein taulukoille lisäsarakeen (**ID**) ja asetin sen **primarykey**ksi, jotta sain vietyä toisten sarakkeiden tyhjät arvot kantaan. Tallennuskoodini näytti nyt tältä (Kuvio 135):

```

Dim UusiRivi As DataRow = Me.KMkanta.TauluPaa.NewRow()

'UusiRivi.Item(0) = TBID.Text
If TBKunta.Text = "" Then MsgBox("Kuntanumero ei voi olla tyhjä! Korjaa arvo ja kokeile uudestaan.", ikkunatyyliioc, "Tyhjä kuntanumero!") : Exit Sub Else UusiRivi.Item(1) = TBKunta.Text
If CBKylat.Text = "" Then MsgBox("Kylävalinta ei voi olla tyhjä! Korjaa arvo ja kokeile uudestaan.", ikkunatyyliioc, "Tyhjä kylävalinta!") : Exit Sub Else UusiRivi.Item(2) = CBKylat.ValueMember
If TBRekMro.Text = "" Then MsgBox("Rekisterinumero ei voi olla tyhjä! Korjaa arvo ja kokeile uudestaan.", ikkunatyyliioc, "Tyhjä rekisterinumero!") : Exit Sub Else UusiRivi.Item(3) = TBRekMro.Text
If TBNimi.Text = "" Then MsgBox("Rakennuksella on oltava nimi! Korjaa arvo ja kokeile uudestaan.", ikkunatyyliioc, "Rakennuksella ei nimeä!") : Exit Sub Else UusiRivi.Item(4) = TBNimi.Text
UusiRivi.Item(5) = TBTontinNimi.Text
If TBKerrosala.Text = "" Then UusiRivi.Item(6) = 0 Else UusiRivi.Item(6) = TBKerrosala.Text
If TBTilavuus.Text = "" Then UusiRivi.Item(7) = 0 Else UusiRivi.Item(7) = TBTilavuus.Text
UusiRivi.Item(8) = CBRakentamivuosi.Text
UusiRivi.Item(9) = CBPeruskorjausvuosi.Text
UusiRivi.Item(10) = CBPaloluokka.Text
UusiRivi.Item(11) = CBKäyttöluokitusryhma.Text
UusiRivi.Item(12) = CBKäyttöluokitus.Text
UusiRivi.Item(13) = CBLammitystapa.Text
UusiRivi.Item(14) = CBAkkitehti.Text
UusiRivi.Item(15) = TBRakennesuunnittelija.Text
UusiRivi.Item(16) = TBLVsuunnittelija.Text
UusiRivi.Item(17) = TBLVsuunnittelija.Text
UusiRivi.Item(18) = TBSahkosuunnittelija.Text
UusiRivi.Item(19) = TBAutomaatiosuunnittelija.Text
UusiRivi.Item(20) = TBErikoissuunnittelija.Text
UusiRivi.Item(21) = TBRakennusurakoitsija.Text
UusiRivi.Item(22) = TBLVurakoitsija.Text
UusiRivi.Item(23) = TBLVurakoitsija.Text
UusiRivi.Item(24) = TBLVurakoitsija.Text
UusiRivi.Item(25) = TBSahkourakoitsija.Text
UusiRivi.Item(26) = TBAutomaatiourakoitsija.Text
UusiRivi.Item(27) = TBErikoisurakoitsija.Text
UusiRivi.Item(28) = TBKiinteistönhoito.Text
UusiRivi.Item(29) = TBLinkki.Text
UusiRivi.Item(30) = TBLuuta.Text

Me.KMkanta.TauluPaa.AddTauluPaaRow(UusiRivi)
Me.TauluPaaTableAdapter.Update(Me.KMkanta.TauluPaa)

```

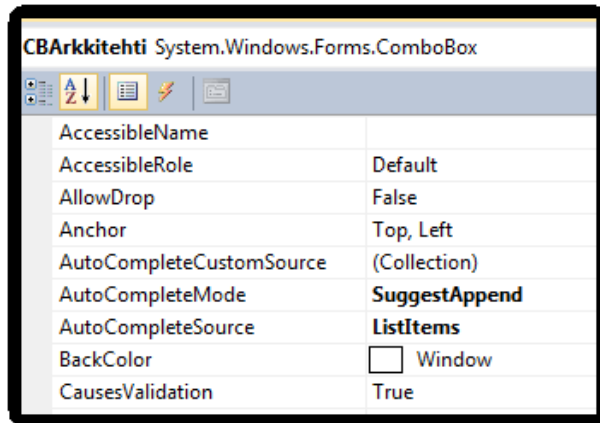
Kuvio 135. Tallennuskoodi rakennuksen tiedoille

Seuraavaksi tein toimivan tallennuspainikkeen, lisäsin vapaalle tekstille kentän, sekä asetin **TabIndex** kohdalleen (Kuvio 136):

Kuvio 136. Valmis syöttölomake

TabIndexeillä määrätään, missä järjestyksessä lomakkeen objekteja edetään tabulaattorilla tai nuolinäppäimillä. Nämä kannattaa määritellä oikein, koska isojen tietomassojen syötössä pitää olla mahdollisuus toimia jouhevasti pelkällä näppäimistöllä.

Sitten aloin miettiä, kuinka saan **Arkkitehti**-kohtaan toimimaan **AutoComplete**-ten. Kuvan mukaisessa tilanteessa lista tulee **TauluArkkitehti** -taulun sisällöstä, eikä päätauluun syötetyistä kuten haluan. (Kuvio 137).



Kuvio 137. AutoComplete

AutoComplete-arvoa ei voi muuttaa **GUI**:n kautta, koska **CustomSource**-valintaa ei voi käyttää, jos **DataSource** on määritetty. Senhän tuli olla **TauluArkkitehti DropDown**ia varten. Totesin, ettei se onnistu kuin koodaamalla. Löysin apua siihen näiltä sivuilta:

<http://net-informations.com/q/faq/autocombo.html>

<http://www.connectionstrings.com/sql-server/>

connectionString vain mietitytti. Huomasin, että voin kopioida sen **app.config**-tiedostosta. Samalla havaitsin, että sieltä sain poistettua kyseisen ylimääräisyyden. Koodi näytti nyt tältä (Kuvio 138):

```

CBArkkitehti.AutoCompleteMode = AutoCompleteMode.Suggest
CBArkkitehti.AutoCompleteSource = AutoCompleteSource.CustomSource
Dim combData As New AutoCompleteStringCollection()
getData(combData)
CBArkkitehti.AutoCompleteCustomSource = combData

End Sub

Private Sub getData(ByVal dataCollection As AutoCompleteStringCollection)
    Dim connectionString As String = Nothing
    Dim connection As SqlConnection
    Dim command As SqlCommand
    Dim adapter As New SqlDataAdapter()
    Dim ds As New DataSet()

    connectionString = "Data Source=POS1230\KIINTEISTOMASSA;Initial Catalog=C:\KANTA\KIINTEISTOMASSAKANTA.MDF;Integrated Security=True"

    Dim sql As String = "SELECT DISTINCT [Arkkitehti] FROM [TauluPaa]"

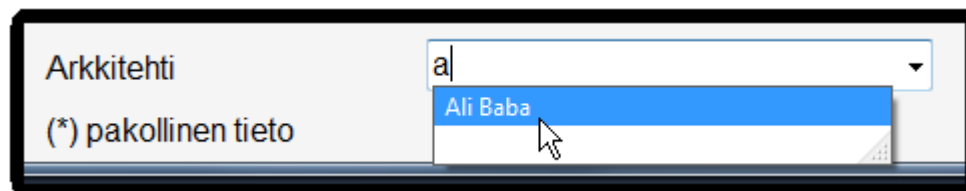
    connection = New SqlConnection(connectionString)

    Try
        connection.Open()
        command = New SqlCommand(sql, connection)
        adapter.SelectCommand = command
        adapter.Fill(ds)
        adapter.Dispose()
        command.Dispose()
        connection.Close()
        For Each row As DataRow In ds.Tables(0).Rows
            dataCollection.Add(row(0).ToString())
        Next
    Catch ex As Exception
        MessageBox.Show("Ei saatu yhteyttä tietokantaan!")
    End Try
End Sub

```

Kuvio 138. AutoCompleten koodi

Koodissa yläosa on **Syotto**-lomakkeen lataamisessa suoritettavassa koodissa ja alhaalla oleva subi omanaan. Testasin toiminnan ja se toimi juuri kuten halusin (Kuvio 139) ja (Kuvio 140).



Kuvio 139. Arkkitehti-kentän automaattitäydennys

Arkkitehti	
Matti Möttönen	a
Matti Möttönen	cc
Ali Baba	jllhj

Kuvio 140. Sarake, josta syöttölomakkeen ehdotukset haetaan

Edellä mainitusta olisi helppo tehdä moduuliin sijoitettava subin sopivin parametrein, jolloin siitä tulisi universaali. Teinkin sen seuraavaksi, koska samaa olisi hyödyllistä käyttää lähes kaikissa loppuissa **Syötto**-lomakkeen kentissä. Moduuliin tein tällaisen subin (Kuvio 141):

```
Sub getData(ByVal dataCollection As AutoCompleteStringCollection, ByVal SarakeNimi As String, ByVal TauluNimi As String)
    Dim connectionString As String = Nothing
    Dim connection As SqlConnection
    Dim command As SqlCommand
    Dim adapter As New SqlDataAdapter()
    Dim ds As New DataSet()

    connectionString = "Data Source=POS1230\KIIINTEISTOMASSA;Initial Catalog=C:\KANTA\KIIINTEISTOMASSAKANTA.MDF;Integrated Security=True"

    Dim sql As String = "SELECT DISTINCT [" & SarakeNimi & "] FROM [" & TauluNimi & "]"

    connection = New SqlConnection(connectionString)

    Try
        connection.Open()
        command = New SqlCommand(sql, connection)
        adapter.SelectCommand = command
        adapter.Fill(ds)
        adapter.Dispose()
        command.Dispose()
        connection.Close()
        For Each row As DataRow In ds.Tables(0).Rows
            dataCollection.Add(row(0).ToString())
        Next
    Catch ex As Exception
        MessageBox.Show("Ei saatu yhteyttä tietokantaan! Tökkäsi sarakeeseen " & SarakeNimi)
    End Try
End Sub
```

Kuvio 141. Subi **AutoCompletea** varten

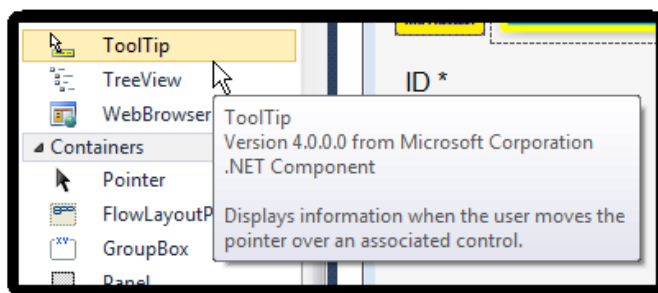
Kuten kuvioista 141 nähdään, asetin parametreiksi sarakkeen, taulun ja hakukoelman. Syöttölomakkeen lataukseen asetin kaikille muillekin kentille kuvan yläosan esimerkin mukaisesti. Nyt sain automaattitäydennyksen toimimaan kaikissa haluamissani kentissä. Se, että kaikkien sarakkeiden tietoja tarjottiin kaikkiin kenttiin, ei haitannut koska samat yritykset tekevät monenlaisia töitä ja olivat siksi relevantteja tarjottavaksi muuallakin.

6.6 Virheilmoitusten suunnittelu

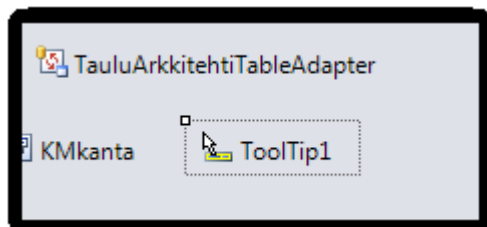
Kuten koodeista näkyy, olin jo toteuttanut hieman virheilmoitusten hallintaa mm. ottamalla huomioon tapaukset, joissa pakollisiin kenttiin ei ole syötetty tietoa. **Msgboxin** parametrin **ikkunatyylilooc** määrittely tulee koodaamistani määrittelyistä moduulista **main**. Lisäsin kuvaavia **messageboxeja** kriittisimpiin sijainteihin ja jäin odottamaan käyttäjien palautetta, mihin niitä mahdollisesti tarvittaisiin lisää.

6.7 *ToolTip*ien lisääminen

ToolTipit, eli työkaluvihjeet, ovat tärkeä osa käyttöliittymän ja ohjelman käyttämisen helpottamista ja omaksumista. Niitä on nopea ja helppo luoda ja niiden avulla saa helposti kerrottua juuri kyseisen kontrollerin käytöstä ilman, että käyttäjän tarvitsisi selata manuaaleja ja etsiä oikeaa ohjetta. Aloitin vihjeiden tekemisen syöttölomakkeesta. Ensin piti ottaa ominaisuus käyttöön (Kuvio 142) ja (Kuvio 143).

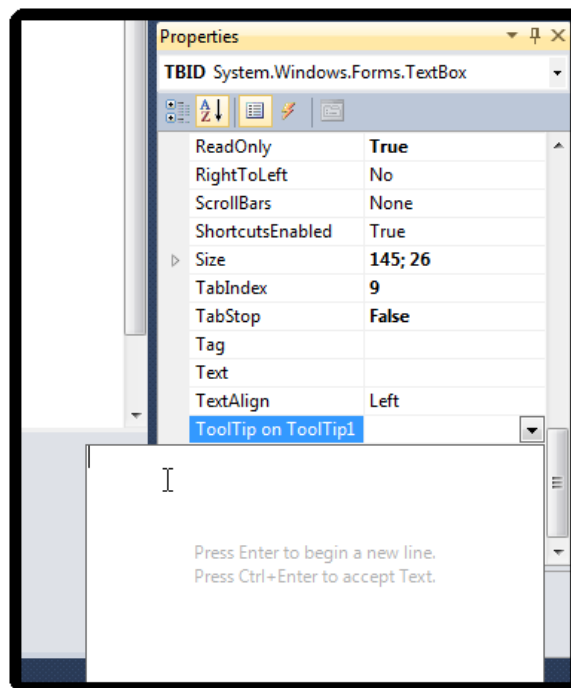


Kuvio 142. Vihjetyökalu työkaluvalinnoissa



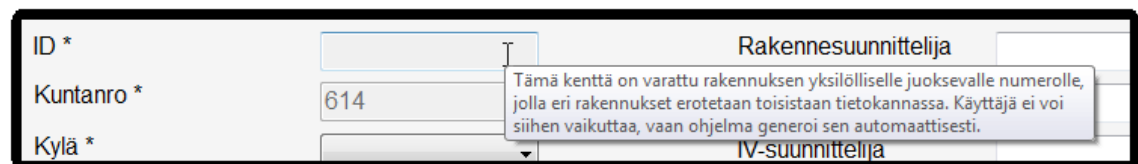
Kuvio 143. Vihjetyökalu otettuna käyttöön

Kun työkalu oli aktivoitu, saattoi aina halutulle objektille asettaa ohjetekstin. Se näytetään aina, kun käyttäjä vie hiiren kyseisen kontrollerin päälle. Aloitin **ID**:n vihjeestä (Kuvio 144).



Kuvio 144. Vihjeen lisääminen kontrollerille

Valmis vihje näkyy ohjelmaa käytettäessä seuraavasti (Kuvio 145):



Kuvio 145. Vihjeen näkyminen käyttäjälle

Jatkoin työtä ja tein **Syotto** -lomakkeen lopuillekin kentille ja painikkeille omat ohjeensa.

7 RAPORTIT, YHTEENVEDOT JA HYÖDYNTÄMINEN

Raportit ohjasin tulostumaan omaan raporttilomakkeeseen. Sinne laitoin vain yhden **DataGridViewin**, jonka lähteenä oli **tauluPaa**. Siellä on myös mahdollista osin poistaa ja muokata tietueita, mutta ei lisätä niitä. Lisääminen tehtäisiin aina syöttölomakkeen kautta, koska valikoiden toiminnallisuuden rakentaminen raporttipuolelle ei ole mielekäästä tehdä eikä käyttää.

Tässä yhteydessä pohdin, kuinka rakentaisin käyttöoikeuksien hallinnan, jotta käyttäjien oikeuksia muuttaa päätaulun tietoja tarvittaessa rajattaisiin. Yhtenä ideana voisi olla, että tarvittavat painikkeet eivät ilmesty ilman erillistä kirjautumista. Tässä vaiheessa ennen kaupallistamista riittää, että minä itse ylläpitäjänä muutan niitä erillisten pyyntöjen perusteella.

8 TULOSTUSTARPEET JA TULOSTUSMODUULIEN KOODAAMINEN

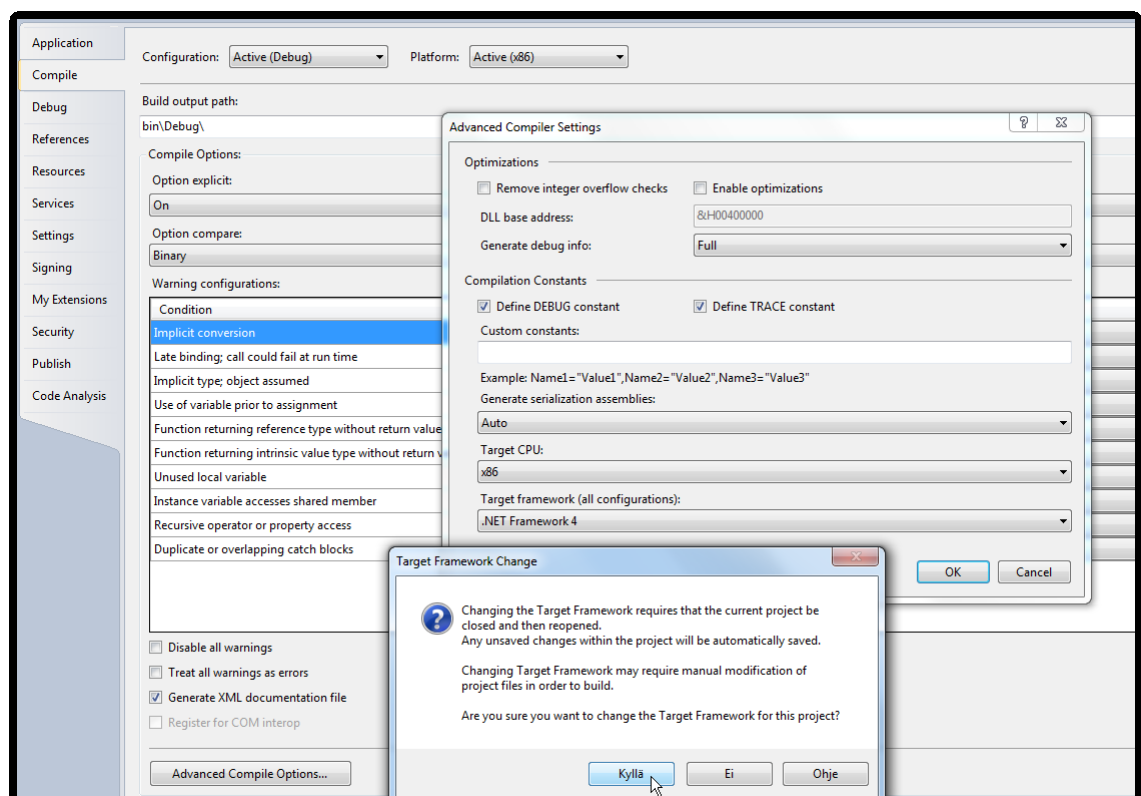
Löysin tällaiset *Programming Tutorialsin* ohjeet, joilla tulostus kannatti suunnitella:

<https://www.youtube.com/watch?v=lfq6nZrR5us>

Latasin ensin *Crystal Reportsin Visual Studiolle* täältä:

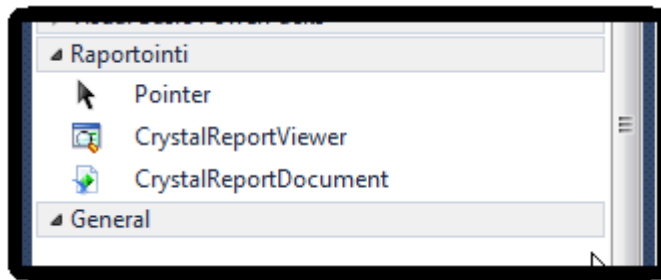
http://downloads.businessobjects.com/akdlm/cr4vs2010/CRforVS_13_0.exe

Asensin sen ja muutin **Compile** asetukseksi **.NET Framework 4** (Kuvio 146).



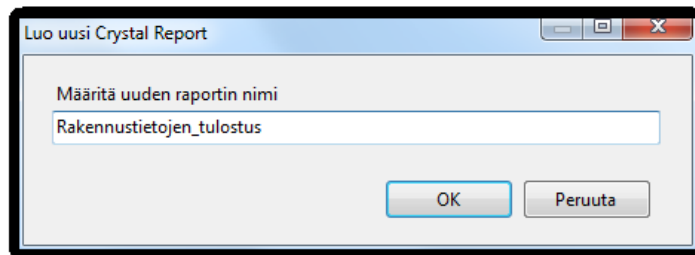
Kuvio 146. **Compile**-asetuksen muuttaminen CR:a varten

Nyt ilmestyi uusia työkaluja, joilla tulostusta voisi suunnitella (Kuvio 147).



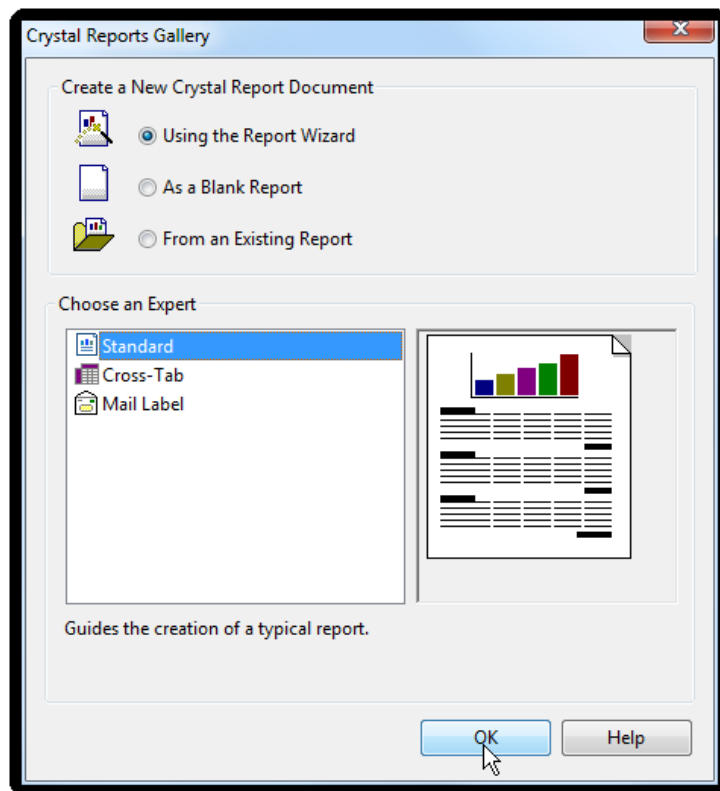
Kuvio 147. *Crystal Report* -työkalut

Sitten aloin tekemään uutta raporttipohjaa ***CrystalReportViewer***-työkalulla, joka käyttäisi suoraan tietokannan taulua/tauluja. Ensimmäinen tehtävä oli antaa pohjalle nimi (Kuvio 148).



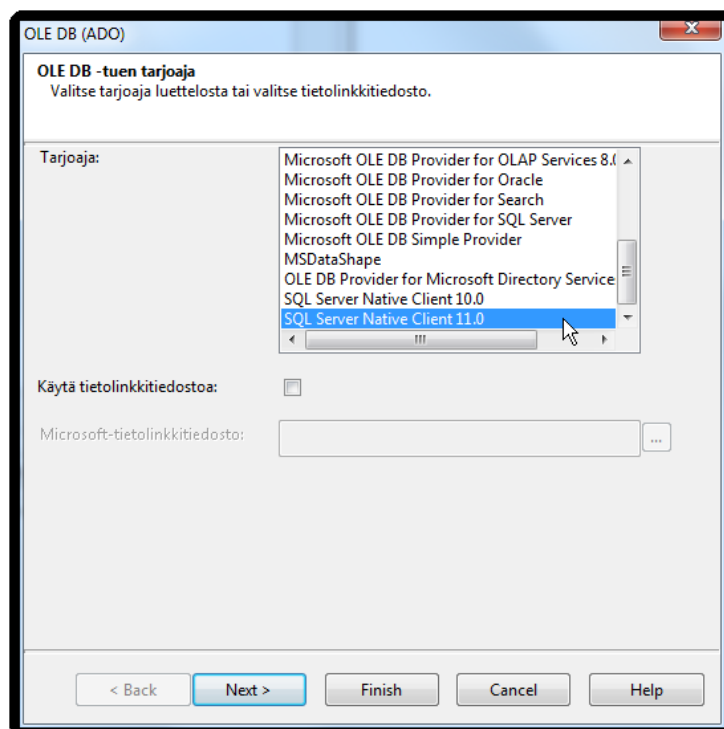
Kuvio 148. Pohjan nimeäminen

Seuraavaksi valitsin ohjatun luonnin (Kuvio 149).



Kuvio 149. Ohjattu pohjan luominen

Sitten valitsin tarjoajan (Kuvio 150).



Kuvio 150. Lähteen valinta

Seuraavaksi lisäsin yhteystiedot (Kuvio 151).

OLE DB (ADO)

Yhteystiedot
Anna tarvittavat tiedot valittuun tietolähteeseen kirjautumista varten.

Palvelin: POS1230\KIINTEISTOMASSA

Käyttäjätunnus:

Salasana:

Tietokanta: C:\KANTA\KIINTEISTOMASSAKANTA.MDF

Integroitu suojaus: ☒

< Back Next > Finish Cancel Help

Kuvio 151. Yhteystietojen asetus

Lisätiedot (menin oletuksilla) (Kuvio 152).

OLE DB (ADO)

Lisätiedot
Päivitä ominaisuuden arvo kaksoisnapsauttamalla ominaisuutta tai valitse ominaisuus ja paina Muuta arvo -painiketta.

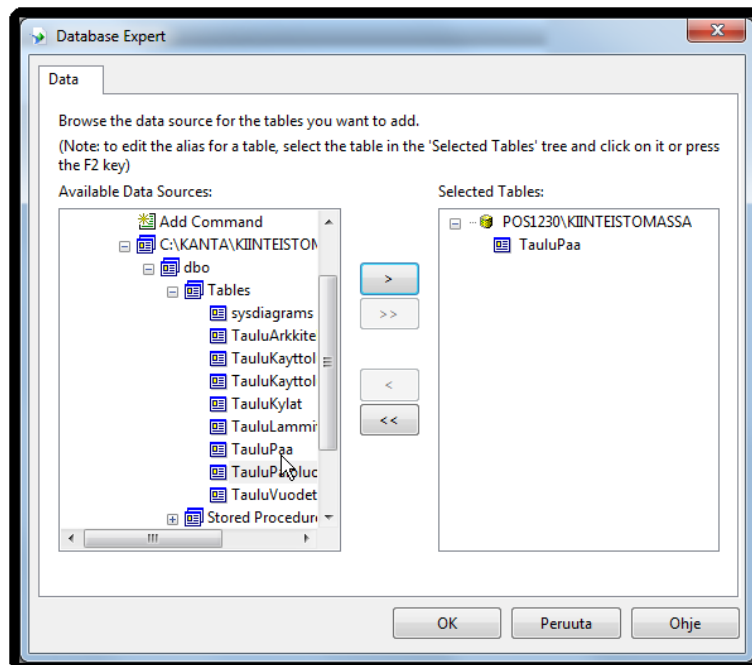
Name	Value
Locale Identifier	1035
Connect Timeout	15
General Timeout	0
OLE DB Services	-5
Current Language	
Auto Translate	-1
Initial File Name	
Use Encryption for Data	0
Replication server name connect o...	
Tag with column collation when p...	0
MARS Connection	0
Failover Partner	
Old Password	

Add Property ... Edit Value ... Remove Property

< Back Next > Finish Cancel Help

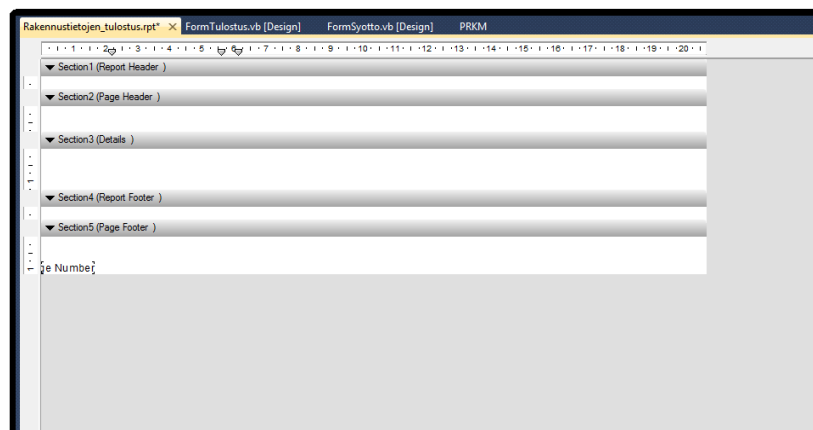
Kuvio 152. Lisätietojen asettaminen

Sitten valitsin taulun, josta tiedot halutaan raporttiin (Kuvio 153).



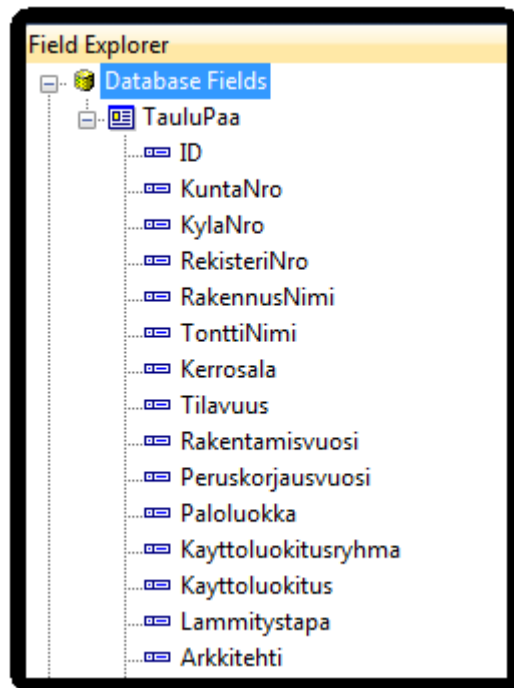
Kuvio 153. Taulun valinta

Nyt oli tyhjä **Crystal Report** -alusta valmis (Kuvio 154).



Kuvio 154. **Crystal Report** -alusta

Raporttipohjaan voidaan nyt lisätä tarvittavia kenttiä suoraan **Field Explorerilla** (Kuvio 155).



Kuvio 155. Lisättävät kentät

Ajattelin tässä vaiheessa, että olisi hyödyllisempää luoda pohja, johon tulostuvat vain ne asiat mitä käyttäjä on suodattanut näkymään **Raportti**-lomakkeessa. Loin pohjan, johon tulostuu raporttilomakkeen **DataGridView**-ohjeitin sisältö painikkeen painalluksella.

9 JOHTOPÄÄTÖKSET

Opinnäytetyö lähti tarpeesta luoda ratkaisu, jolla Posion kunnan omistamien kiinteistöjen tiedot saataisiin keskitetyksi yhteen paikkaan ja jossa niitä olisi helppo ylläpitää ja hallinnoida. Tähän katsoin parhaaksi ratkaisuksi niin kustannusten kuin käytettävyyden ja skaalautuvuudenkin osalta tietokantapalvelimen/tietokannan rakentamisen sekä helppokäyttöisen ja kompaktin erillisen käyttöliittymän luomisen sen ylläpitoon ja käyttöön.

Rakensin tietokantapalvelimeksi *MS SQL Server 2012 SP1*:n, joka toimii tunnetusti hyvin *Windows AD* -ympäristössä ja loin siihen kannan, johon tiedot vietäisiin ja jonka käyttäjiksi valikoitui *tekninen AD*-ryhmä. Käyttöliittymä rakentui pitkälti käyttäjien tarpeiden pohjalta ja osa valikkorakenteesta tulee suoraan RT-kortistosta.

Erilaiset tietokannat ovat yrityksissä laajalti käytössä, mutta niiden osaamista harvemmin löytyy paikan päältä saati, että niitä rakennettaisiin itse. Lisenssit ovat kaupallisilla ratkaisuilla usein erittäin kalliita (mm. *Oracle*) ja ylläpito kaukana ja kallista. Siksi niitä yleensä käytetäänkin vain erittäin perustellusta syytä ja isommissa kokonaisuuksissa. Tekemäni ohjelmisto ja tietokanta mahdollistivat tällaisen pienemmänkin tarpeen paikkaamisen kokonaisvaltaisella tietokantaratkaisulla. Lopputuloksena saatiin keskitetty, helposti ylläpidettävä tietokanta ja käyttöliittymä, joka on ylläpitäjän helposti muokattavissa sekä kevyt jaella.

Tapani tehdä opinnäytetyö tarkasti ja laajasti dokumentoiden toimii myöhemmin myös hyvänä oppaana sille, kuinka teen vastaavan ohjelmiston johonkin toiseen tarkoitukseen. Oppaan seuraava käyttökohde on jo selvillä ja se tulee olemaan terveyskeskuksen varastokirjanpito-ohjelmiston uudistaminen tietokantaympäristöön.

LÄHTEET

Microsoft 2015a. Asennuspaketit MS SQL Server 2008 R2 Viitattu 3.12.2015
<http://www.microsoft.com/en-us/download/details.aspx?id=30438>

- 2015b. Asennuspaketti MS SQL Server 2008 R2 XP:lle Viitattu 3.12.2015
<http://www.microsoft.com/fi-fi/download/details.aspx?id=8483>

- 2015c. Ohje MS SQL Server 2008 R2 asennuksen suorittamiseen Viitattu 3.12.2015
<http://blogs.msdn.com/b/petersad/archive/2009/11/13/how-to-install-sql-server-2008-r2-express-edition-november-ctp.aspx>

- 2015d. Management Studion asennuspaketti Viitattu 3.12.2015
<https://www.microsoft.com/en-us/download/details.aspx?id=7593>

- 2015e. Microsoft Visual Studio 2010 Expressin asennuspaketti Viitattu 3.12.2015
<http://www.microsoft.com/visualstudio/eng/products/visual-studio-2010-express>

- 2015f. Tietokannan luominen Viitattu 3.12.2015 <http://msdn.microsoft.com/en-us/library/ms233763%28v=vs.90%29.aspx>

- 2015g. Ohje verkkoasetuksiin Viitattu 3.12.2015
[http://technet.microsoft.com/en-us/library/ms189083\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms189083(v=sql.105).aspx)

- 2015h. Microsoft SQL Server 2008 Service Pack 3 Feature Pack latauslinkki Viitattu 3.12.2015
<http://www.microsoft.com/en-us/download/details.aspx?id=27596>

- 2015i. Protokollien aktivointi Viitattu 3.12.2015 [http://technet.microsoft.com/en-us/library/ms345343\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms345343(v=sql.105).aspx)

- 2015j. Protokollien aktivointi Viitattu 3.12.2015 [http://technet.microsoft.com/en-us/library/ms181087\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms181087(v=sql.105).aspx)

- 2015k. Tietokannan tunnushallinta Viitattu 3.12.2015
<http://technet.microsoft.com/en-us/library/aa337545.aspx>

- 2015l. Tietokannan tunnushallinta Viitattu 3.12.2015
<http://msdn.microsoft.com/en-us/library/aa337562.aspx>

- 2015m. Microsoft® SQL Server® 2012 Express latauslinkki Viitattu 3.12.2015
<https://www.microsoft.com/en-us/download/details.aspx?id=29062>

- 2015n. Asennusohje Microsoft® SQL Server® 2012 Viitattu 3.12.2015
<https://technet.microsoft.com/en-us/library/bb500469%28v=sql.110%29.aspx>

- 2015o. Asennusohje Microsoft® SQL Server® 2013 Viitattu 3.12.2015
<https://technet.microsoft.com/en-us/library/ms143219%28v=sql.110%29.aspx>

- 2015p. Ohje paikalliseen tietokantaan yhdistämisestä Viitattu 3.12.2015
[http://technet.microsoft.com/en-us/library/ms345332\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms345332(v=sql.105).aspx)

- 2015q. Ohje paikalliseen tietokantaan yhdistämisestä Viitattu 3.12.2015
<http://msdn.microsoft.com/en-us/library/vstudio/ms239722.aspx>

- 2015r. Clientin verkkoasetukset Viitattu 3.12.2015
[http://technet.microsoft.com/en-us/library/ms190611\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms190611(v=sql.105).aspx)

- 2015s. Client tietokantayhteys Viitattu 3.12.2015
[http://technet.microsoft.com/en-us/library/ms188691\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms188691(v=sql.105).aspx)

- 2015t. ODBC-yhteyden luominen Viitattu 3.12.2015
<http://support.microsoft.com/kb/310985>

- 2015u. Visual Studio Ultimaten Designer-ohje Viitattu 3.12.2015
<http://msdn.microsoft.com/en-us/library/ms233763%28v=vs.90%29.aspx>

- 2015v. BindingNavigator-ohjausobjektin luominen ja tallennustoiminto Viitattu 3.12.2015
<http://msdn.microsoft.com/en-us/library/2b4be09b%28v=vs.90%29.aspx>

- 2015w. BindingNavigator-ohjausobjektin luominen ja tallennustoiminto Viitattu 3.12.2015
<http://msdn.microsoft.com/en-us/library/8zhc8d2f%28v=vs.85%29.aspx>

- 2015x. BindingNavigator-ohjausobjektin luominen ja tallennustoiminto Viitattu 3.12.2015
[https://msdn.microsoft.com/en-us/library/safa4957\(v=vs.80\).aspx](https://msdn.microsoft.com/en-us/library/safa4957(v=vs.80).aspx)

Stackoverflow 2015. Tietokantataulun pääavaimen automaattinen numerointi Viitattu 3.12.2015
<http://stackoverflow.com/questions/10991894/auto-increment-primary-key-in-sql-server-management-studio-2012>

Programming Tutorials 2015. Tietokantataulun pääavaimen automaattinen numerointi. Viitattu 3.12.2015
<https://www.youtube.com/watch?v=lfq6nZrR5us>

Wikipedia 2015. Visual Studio Ultimatesta Viitattu 3.12.2015
http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#Visual_Studio_Ultimate

Home And Learn 2015. Client tietokantayhteys Viitattu 3.12.2015
<http://www.homeandlearn.co.uk/NET/nets12p4.html>

LIITTEET

Liite 1. Excel-tiedosto Tietokantaa.xls